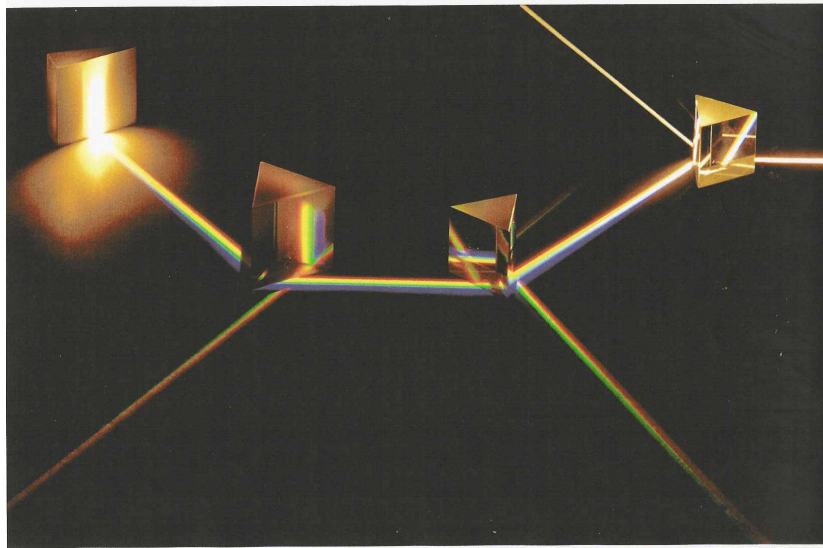


# Fourier and Wavelet Signal Processing



Martin Vetterli

*École Polytechnique Fédérale de Lausanne and University of California, Berkeley*

Jelena Kovačević

*Carnegie Mellon University*

Vivek K Goyal

*Massachusetts Institute of Technology*

October 30, 2011

Copyright (c) 2011 Martin Vetterli, Jelena Kovačević, and Vivek K Goyal.

These materials are protected by copyright under the  
Attribution-NonCommercial-NoDerivs 3.0 Unported License  
from Creative Commons.

Cover photograph by Christiane Grimm, Geneva, Switzerland.

Experimental set up by Prof. Libero Zuppiroli and Philippe Bugnon, Laboratory of Optoelectronics Molecular Materials, EPFL, Lausanne, Switzerland.

The photograph captures an experiment first described by Isaac Newton in *Opticks* in 1730, explaining how the white light can be split into its color components and then resynthesized. It is a physical implementation of a white light decomposition into its Fourier components—the colors of the rainbow, followed by a synthesis to recover the original. This experiment graphically summarizes the major theme of the book—many signals can be split into essential components, and the signal's characteristics can be better understood by looking at its components; the process called *analysis*. These components can be used for processing the signal; more importantly, the signal can be perfectly recovered from those same components, through the process called *synthesis*.

# Contents

<b>Image Attribution</b>	<b>xiii</b>
<b>Quick Reference</b>	<b>xv</b>
<b>Preface</b>	<b>xxi</b>
<b>Release Notes</b>	<b>xxv</b>
<b>Acknowledgments</b>	<b>xxvii</b>
<b>From Rainbows to Spectra</b>	<b>1</b>
<b>Part I: Foundations of Signals and Systems</b>	<b>7</b>
<b>1 From Euclid to Hilbert</b>	<b>9</b>
1.1 Introduction . . . . .	10
1.2 Vector Spaces . . . . .	18
1.2.1 Definition and Properties . . . . .	18
1.2.2 Inner Product . . . . .	22
1.2.3 Norm . . . . .	25
1.2.4 Standard Spaces . . . . .	28
1.3 Hilbert Spaces . . . . .	34
1.3.1 Convergence . . . . .	34
1.3.2 Completeness . . . . .	36
1.3.3 Linear Operators . . . . .	38
1.4 Approximations, Projections, and Decompositions . . . . .	47
1.4.1 Projection Theorem . . . . .	48
1.4.2 Projection Operators . . . . .	52
1.4.3 Direct Sums and Subspace Decompositions . . . . .	56
1.4.4 Minimum Mean-Squared Error Estimation . . . . .	59
1.5 Bases and Frames . . . . .	65
1.5.1 Bases and Riesz Bases . . . . .	65
1.5.2 Orthonormal Bases . . . . .	70
1.5.3 Biorthogonal Pairs of Bases . . . . .	80
1.5.4 Frames . . . . .	95

1.5.5	Matrix Representations of Vectors and Linear Operators . . . . .	103
1.6	Computational Aspects . . . . .	113
1.6.1	Cost, Complexity, and Asymptotic Notations . . . . .	114
1.6.2	Precision . . . . .	117
1.6.3	Conditioning . . . . .	120
1.6.4	Solving Systems of Linear Equations . . . . .	122
1.A	Elements of Analysis and Topology . . . . .	128
1.A.1	Basic Definitions . . . . .	128
1.A.2	Convergence . . . . .	129
1.A.3	Interchange Theorems . . . . .	131
1.A.4	Inequalities . . . . .	132
1.A.5	Integration by Parts . . . . .	133
1.B	Elements of Linear Algebra . . . . .	133
1.B.1	Basic Definitions and Properties . . . . .	134
1.B.2	Special Matrices . . . . .	140
1.C	Elements of Probability . . . . .	144
1.C.1	Basic Definitions . . . . .	144
1.C.2	Standard Distributions . . . . .	146
1.C.3	Estimation . . . . .	148
	Chapter at a Glance . . . . .	151
	Historical Remarks . . . . .	152
	Further Reading . . . . .	152
	Exercises with Solutions . . . . .	153
	Exercises . . . . .	157
<b>2</b>	<b>Sequences and Discrete-Time Systems</b>	<b>171</b>
2.1	Introduction . . . . .	172
2.2	Sequences . . . . .	175
2.2.1	Infinite-Length Sequences . . . . .	175
2.2.2	Finite-Length Sequences . . . . .	182
2.2.3	Multidimensional Sequences . . . . .	183
2.3	Systems . . . . .	185
2.3.1	Discrete-Time Systems and Their Properties . . . . .	185
2.3.2	Difference Equations . . . . .	193
2.3.3	Linear Shift-Invariant Systems . . . . .	194
2.4	Discrete-Time Fourier Transform . . . . .	205
2.4.1	Definition of the DTFT . . . . .	205
2.4.2	Existence and Convergence of the DTFT . . . . .	207
2.4.3	Properties of the DTFT . . . . .	210
2.4.4	Frequency Response of Filters . . . . .	216
2.5	$z$ -Transform . . . . .	223
2.5.1	Definition of the $z$ -Transform . . . . .	223
2.5.2	Existence and Convergence of the $z$ -Transform . . . . .	225
2.5.3	Properties of the $z$ -Transform . . . . .	230
2.5.4	$z$ -Transform of Filters . . . . .	237



Contents	v
2.6 Discrete Fourier Transform . . . . .	240
2.6.1 Definition of the DFT . . . . .	241
2.6.2 Properties of the DFT . . . . .	244
2.6.3 Frequency Response of Filters . . . . .	247
2.7 Multirate Sequences and Systems . . . . .	248
2.7.1 Downsampling . . . . .	249
2.7.2 Upsampling . . . . .	252
2.7.3 Downsampling and Upsampling . . . . .	253
2.7.4 Downsampling, Upsampling and Filtering . . . . .	254
2.7.5 Multirate Identities . . . . .	259
2.7.6 Polyphase Representation . . . . .	261
2.8 Discrete Stochastic Processes and Systems . . . . .	268
2.8.1 Processes . . . . .	268
2.8.2 Systems . . . . .	269
2.8.3 Discrete-Time Fourier Transform . . . . .	271
2.8.4 Multirate Sequences and Systems . . . . .	275
2.9 Computational Aspects . . . . .	281
2.9.1 Fast Fourier Transforms . . . . .	282
2.9.2 Convolution . . . . .	285
2.9.3 Multirate Operations . . . . .	288
2.A Elements of Analysis . . . . .	290
2.A.1 Complex Numbers . . . . .	290
2.A.2 Difference Equations . . . . .	292
2.A.3 Convergence of the Convolution Sum . . . . .	293
2.B Elements of Algebra . . . . .	293
2.B.1 Polynomials . . . . .	293
2.B.2 Vectors and Matrices of Polynomials . . . . .	296
2.B.3 Kronecker Product . . . . .	299
Chapter at a Glance . . . . .	300
Historical Remarks . . . . .	303
Further Reading . . . . .	303
Exercises with Solutions . . . . .	304
Exercises . . . . .	310
<b>3 Functions and Continuous-Time Systems</b>	<b>317</b>
3.1 Introduction . . . . .	318
3.2 Functions . . . . .	319
3.2.1 Functions on the Real Line . . . . .	319
3.2.2 Periodic Functions . . . . .	326
3.2.3 Multidimensional Functions . . . . .	326
3.3 Systems . . . . .	327
3.3.1 Continuous-Time Systems and Their Properties . . . . .	327
3.3.2 Differential Equations . . . . .	330
3.3.3 Linear Shift-Invariant Systems . . . . .	330
3.4 Fourier Transform . . . . .	334
3.4.1 Definition of the Fourier Transform . . . . .	335

3.4.2	Existence and Convergence of the Fourier Transform	336
3.4.3	Properties of the Fourier Transform . . . . .	342
3.4.4	Frequency Response of Filters . . . . .	353
3.4.5	Laplace Transform . . . . .	353
3.5	Fourier Series . . . . .	355
3.5.1	Definition of the Fourier Series . . . . .	355
3.5.2	Existence of the Fourier Series . . . . .	359
3.5.3	Properties of the Fourier Series . . . . .	359
3.5.4	Frequency Response of Filters . . . . .	363
3.6	Continuous Stochastic Processes and Systems . . . . .	363
3.6.1	Processes . . . . .	364
3.6.2	Systems . . . . .	365
3.6.3	Fourier Transform . . . . .	366
	Chapter at a Glance . . . . .	368
	Historical Remarks . . . . .	369
	Further Reading . . . . .	370
	Exercises with Solutions . . . . .	370
	Exercises . . . . .	372
<b>4</b>	<b>Sampling and Interpolation</b>	<b>377</b>
4.1	Introduction . . . . .	378
4.2	Finite-Dimensional Vectors . . . . .	385
4.2.1	Sampling and Interpolation with Orthonormal Vectors	385
4.2.2	Sampling and Interpolation with Nonorthogonal Vectors . . . . .	389
4.3	Sequences . . . . .	393
4.3.1	Sampling and Interpolation with Orthonormal Sequences . . . . .	394
4.3.2	Sampling and Interpolation for Bandlimited Sequences	398
4.3.3	Sampling and Interpolation with Nonorthogonal Sequences . . . . .	401
4.4	Functions . . . . .	404
4.4.1	Sampling and Interpolation with Orthonormal Functions . . . . .	405
4.4.2	Sampling and Interpolation for Bandlimited Functions	408
4.4.3	Sampling and Interpolation with Nonorthogonal Functions . . . . .	422
4.5	Periodic Functions . . . . .	426
4.5.1	Sampling and Interpolation with Orthonormal Periodic Functions . . . . .	428
4.5.2	Sampling and Interpolation for Bandlimited Periodic Functions . . . . .	430
4.6	Stochastic Vectors and Processes . . . . .	436
4.6.1	Finite-Dimensional Stochastic Vectors . . . . .	437
4.6.2	Discrete Bandlimited Stochastic Processes . . . . .	437
4.6.3	Continuous Bandlimited Stochastic Processes . . . . .	440

Contents	vii
4.7 Computational Aspects . . . . .	440
4.7.1 Projection Onto Convex Sets . . . . .	440
Chapter at a Glance . . . . .	445
Historical Remarks . . . . .	447
Further Reading . . . . .	447
Exercises with Solutions . . . . .	448
Exercises . . . . .	452
<b>5 Approximation and Compression</b>	<b>455</b>
5.1 Introduction . . . . .	456
5.2 Approximation of Functions on Finite Intervals by Polynomials . . . . .	461
5.2.1 Least-Squares Approximation . . . . .	462
5.2.2 Lagrange Interpolation: Matching Points . . . . .	465
5.2.3 Taylor Series Expansion: Matching Derivatives . . . . .	468
5.2.4 Hermite Interpolation: Matching Points and Derivatives . . . . .	470
5.2.5 Minimax Polynomial Approximation . . . . .	471
5.3 Approximation of Functions by Splines . . . . .	480
5.3.1 Approximation in Shift-Invariant Subspaces Using Splines . . . . .	481
5.3.2 Approximation in Shift-Invariant Subspaces Using Orthogonalized Splines . . . . .	483
5.3.3 Continuous-Time Processing Using Discrete-Time Operators in Spline Spaces . . . . .	485
5.3.4 Polynomial Reproduction and Strang–Fix Theorem . . . . .	490
5.4 Approximation of Functions and Sequences by Series Truncation . . . . .	493
5.4.1 Linear Approximation . . . . .	494
5.4.2 Nonlinear Approximation . . . . .	494
5.4.3 Approximation in Fourier Bases . . . . .	496
5.4.4 Karhunen–Loève Transform . . . . .	497
5.5 Compression and Transform Coding . . . . .	500
5.5.1 Transform Coding . . . . .	503
5.5.2 Optimal Transforms for Transform Coding . . . . .	510
5.6 Computational Aspects . . . . .	513
5.6.1 Optimal Quantization and Clustering . . . . .	513
5.6.2 Projection Onto Convex Sets . . . . .	513
Chapter at a Glance . . . . .	514
Historical Remarks . . . . .	515
Further Reading . . . . .	516
Exercises with Solutions . . . . .	516
Exercises . . . . .	520
<b>6 Time-Frequency Localization</b>	<b>525</b>
6.1 Introduction . . . . .	526
6.2 Localization for Functions . . . . .	528
6.2.1 Time Localization . . . . .	528

6.2.2	Frequency Localization . . . . .	530
6.2.3	Uncertainty Principle for Functions . . . . .	531
6.2.4	Scale Localization . . . . .	535
6.3	Localization for Sequences . . . . .	536
6.3.1	Time Localization . . . . .	536
6.3.2	Frequency Localization . . . . .	538
6.3.3	Uncertainty Principle for Sequences . . . . .	539
6.3.4	Scale Localization . . . . .	542
6.3.5	Uncertainty Principle for Finite-Length Sequences . . . . .	543
	Chapter at a Glance . . . . .	545
	Historical Remarks . . . . .	546
	Further Reading . . . . .	546
	Exercises with Solutions . . . . .	546
	Exercises . . . . .	550
<b>Intermezzo</b>		<b>553</b>
<b>Part II: Structured Representations for Signal Processing</b>		<b>565</b>
<b>7</b>	<b>Filter Banks: Building Blocks of Time-Frequency Expansions</b>	<b>567</b>
7.1	Introduction . . . . .	568
7.2	Orthogonal Two-Channel Filter Banks . . . . .	572
7.2.1	A Single Channel and Its Properties . . . . .	573
7.2.2	Complementary Channels and Their Properties . . . . .	576
7.2.3	Orthogonal Two-Channel Filter Bank . . . . .	577
7.2.4	Polyphase View of Orthogonal Filter Banks . . . . .	580
7.2.5	Polynomial Approximation by Filter Banks . . . . .	584
7.3	Design of Orthogonal Two-Channel Filter Banks . . . . .	585
7.3.1	Lowpass Approximation Design . . . . .	586
7.3.2	Polynomial Approximation Design . . . . .	588
7.3.3	Lattice Factorization Design . . . . .	590
7.4	Biorthogonal Two-Channel Filter Banks . . . . .	592
7.4.1	A Single Channel and Its Properties . . . . .	594
7.4.2	Complementary Channels and Their Properties . . . . .	597
7.4.3	Biorthogonal Two-Channel Filter Bank . . . . .	598
7.4.4	Polyphase View of Biorthogonal Filter Banks . . . . .	600
7.4.5	Linear-Phase Two-Channel Filter Banks . . . . .	601
7.5	Design of Biorthogonal Two-Channel Filter Banks . . . . .	602
7.5.1	Factorization Design . . . . .	603
7.5.2	Complementary Filter Design . . . . .	604
7.5.3	Lifting Design . . . . .	605
7.6	Two-Channel Filter Banks with Stochastic Inputs . . . . .	607
7.7	Computational Aspects . . . . .	608
7.7.1	Two-Channel Filter Banks . . . . .	608
7.7.2	Boundary Extensions . . . . .	611
	Chapter at a Glance . . . . .	615

Historical Remarks . . . . .	618
Further Reading . . . . .	618
Exercises with Solutions . . . . .	620
Exercises . . . . .	625
<b>8 Local Fourier Bases on Sequences</b>	<b>631</b>
8.1 Introduction . . . . .	632
8.2 $N$ -Channel Filter Banks . . . . .	635
8.2.1 Orthogonal $N$ -Channel Filter Banks . . . . .	635
8.2.2 Polyphase View of $N$ -Channel Filter Banks . . . . .	637
8.3 Complex Exponential-Modulated Local Fourier Bases . . . . .	642
8.3.1 Balian-Low Theorem . . . . .	643
8.3.2 Application to Power Spectral Density Estimation . . . . .	644
8.3.3 Application to Communications . . . . .	649
8.4 Cosine-Modulated Local Fourier Bases . . . . .	651
8.4.1 Lapped Orthogonal Transforms . . . . .	652
8.4.2 Application to Audio Compression . . . . .	659
8.5 Computational Aspects . . . . .	661
Chapter at a Glance . . . . .	663
Historical Remarks . . . . .	666
Further Reading . . . . .	666
Exercises with Solutions . . . . .	666
Exercises . . . . .	671
<b>9 Wavelet Bases on Sequences</b>	<b>675</b>
9.1 Introduction . . . . .	676
9.2 Tree-Structured Filter Banks . . . . .	683
9.2.1 The Lowpass Channel and Its Properties . . . . .	683
9.2.2 Bandpass Channels and Their Properties . . . . .	687
9.2.3 Relationship between Lowpass and Bandpass Channels . . . . .	689
9.3 Orthogonal Discrete Wavelet Transform . . . . .	690
9.3.1 Definition of the Orthogonal DWT . . . . .	690
9.3.2 Properties of the Orthogonal DWT . . . . .	691
9.4 Biorthogonal Discrete Wavelet Transform . . . . .	696
9.4.1 Definition of the Biorthogonal DWT . . . . .	696
9.4.2 Properties of the Biorthogonal DWT . . . . .	698
9.5 Wavelet Packets . . . . .	698
9.5.1 Definition of the Wavelet Packets . . . . .	699
9.5.2 Properties of the Wavelet Packets . . . . .	700
9.6 Computational Aspects . . . . .	700
Chapter at a Glance . . . . .	702
Historical Remarks . . . . .	703
Further Reading . . . . .	703
Exercises with Solutions . . . . .	703
9.7 Introduction . . . . .	703
Exercises . . . . .	710

x		Contents
9.8	Introduction . . . . .	710
<b>10</b>	<b>Local Fourier and Wavelet Frames on Sequences</b>	<b>713</b>
10.1	Introduction . . . . .	714
10.2	Finite-Dimensional Frames . . . . .	725
10.2.1	Tight Frames for $\mathbb{C}^N$ . . . . .	725
10.2.2	General Frames for $\mathbb{C}^N$ . . . . .	732
10.2.3	Choosing the Expansion Coefficients . . . . .	737
10.3	Oversampled Filter Banks . . . . .	744
10.3.1	Tight Oversampled Filter Banks . . . . .	744
10.3.2	Polyphase View of Oversampled Filter Banks . . . . .	747
10.4	Local Fourier Frames . . . . .	750
10.4.1	Complex Exponential-Modulated Local Fourier Frames	751
10.4.2	Cosine-Modulated Local Fourier Frames . . . . .	754
10.5	Wavelet Frames . . . . .	757
10.5.1	Oversampled DWT . . . . .	757
10.5.2	Pyramid Frames . . . . .	759
10.5.3	Shift-Invariant DWT . . . . .	762
10.6	Computational Aspects . . . . .	763
10.6.1	The Algorithm à Trous . . . . .	763
10.6.2	Efficient Gabor and Spectrum Computation . . . . .	764
10.6.3	Efficient Sparse Frame Expansions . . . . .	764
	Chapter at a Glance . . . . .	765
	Historical Remarks . . . . .	766
	Further Reading . . . . .	766
	Exercises with Solutions . . . . .	768
	Exercises . . . . .	772
<b>11</b>	<b>Local Fourier Transforms, Frames and Bases on Functions</b>	<b>775</b>
11.1	Introduction . . . . .	775
11.2	Local Fourier Transform . . . . .	776
11.2.1	Definition of the Local Fourier Transform . . . . .	776
11.2.2	Properties of the Local Fourier Transform . . . . .	779
11.3	Local Fourier Frame Series . . . . .	785
11.3.1	Sampling Grids . . . . .	785
11.3.2	Frames from Sampled Local Fourier Transform . . . . .	785
11.4	Local Fourier Series . . . . .	785
11.4.1	Complex Exponential-Modulated Local Fourier Bases	785
11.4.2	Cosine-Modulated Local Fourier Bases . . . . .	785
11.5	Computational Aspects . . . . .	786
11.5.1	Complex Exponential-Modulated Local Fourier Bases	786
11.5.2	Cosine-Modulated Local Fourier Bases . . . . .	786
	Chapter at a Glance . . . . .	786
	Historical Remarks . . . . .	786
	Further Reading . . . . .	786
	Exercises with Solutions . . . . .	786

Contents	xi
Exercises . . . . .	786
<b>12 Wavelet Bases, Frames and Transforms on Functions</b>	<b>787</b>
12.1 Introduction . . . . .	787
12.1.1 Scaling Function and Wavelets from Haar Filter Bank	788
12.1.2 Haar Wavelet Series . . . . .	793
12.1.3 Haar Frame Series . . . . .	800
12.1.4 Haar Continuous Wavelet Transform . . . . .	802
12.2 Scaling Function and Wavelets from Orthogonal Filter Banks .	806
12.2.1 Iterated Filters . . . . .	806
12.2.2 Scaling Function and its Properties . . . . .	807
12.2.3 Wavelet Function and its Properties . . . . .	816
12.2.4 Scaling Function and Wavelets from Biorthogonal Filter Banks . . . . .	818
12.3 Wavelet Series . . . . .	820
12.3.1 Definition of the Wavelet Series . . . . .	821
12.3.2 Properties of the Wavelet Series . . . . .	825
12.3.3 Multiresolution Analysis . . . . .	828
12.3.4 Biorthogonal Wavelet Series . . . . .	837
12.4 Wavelet Frame Series . . . . .	840
12.4.1 Definition of the Wavelet Frame Series . . . . .	840
12.4.2 Frames from Sampled Wavelet Series . . . . .	840
12.5 Continuous Wavelet Transform . . . . .	840
12.5.1 Definition of the Continuous Wavelet Transform .	840
12.5.2 Existence and Convergence of the Continuous Wavelet Transform . . . . .	841
12.5.3 Properties of the Continuous Wavelet Transform .	842
12.6 Computational Aspects . . . . .	852
12.6.1 Wavelet Series: Mallat's Algorithm . . . . .	852
12.6.2 Wavelet Frames . . . . .	856
Chapter at a Glance . . . . .	856
Historical Remarks . . . . .	856
Further Reading . . . . .	857
Exercises with Solutions . . . . .	857
Exercises . . . . .	859
<b>13 Approximation, Estimation, and Compression</b>	<b>865</b>
13.1 Introduction . . . . .	866
13.2 Abstract Models and Approximation . . . . .	866
13.2.1 Local Fourier and Wavelet Approximations of Piece- wise Smooth Functions . . . . .	866
13.2.2 Wide-Sense Stationary Gaussian Processes . . . . .	866
13.2.3 Poisson Processes . . . . .	866
13.3 Empirical Models . . . . .	866
13.3.1 $\ell^p$ Models . . . . .	866
13.3.2 Statistical Models . . . . .	866

---

13.4	Estimation and Denoising . . . . .	866
13.4.1	Connections to Approximation . . . . .	866
13.4.2	Wavelet Thresholding and Variants . . . . .	866
13.4.3	Frames . . . . .	866
13.5	Compression . . . . .	866
13.5.1	Audio Compression . . . . .	866
13.5.2	Image Compression . . . . .	866
13.6	Inverse Problems . . . . .	866
13.6.1	Deconvolution . . . . .	866
13.6.2	Compressed Sensing . . . . .	866
	Chapter at a Glance . . . . .	866
	Historical Remarks . . . . .	866
	Further Reading . . . . .	866
13.A	Elements of Source Coding . . . . .	867
13.A.1	Entropy Coding . . . . .	867
13.A.2	Quantization . . . . .	867
13.A.3	Transform Coding . . . . .	867

**Bibliography****869**



# Image Attribution

Source	Permission
<sup>s1</sup> Christiane Grimm, Geneva, Switzerland	<sup>p1</sup> Permission of the copyright holder
<sup>s2</sup> Wikimedia Commons	<sup>p2</sup> Copyright expired
<sup>s3</sup> Nobelprize.org	<sup>p3</sup> GNU Free Documentation license
<sup>s4</sup> KerryR.net	<sup>p4</sup> Public domain
<sup>s5</sup> Edward Lee's home page	<sup>p5</sup> Common property, no original authorship
	<sup>p6</sup> NASA material, not protected by copyright
	<sup>p7</sup> Public domain in the US
	<sup>p8</sup> Unknown rights

## Front Material

- *Cover photograph*.<sup>s1,p1</sup> Experimental set up by Prof. Libero Zuppiroli, Laboratory of Optoelectronics Molecular Materials, EPFL, Lausanne, Switzerland.

## Chapter 0

- *Rainbow explained*.<sup>s2,p2</sup>

## Chapter 1

- *Euclid of Megara*.<sup>s2,p2</sup> Panel from the Series “Famous Men,” Justus of Ghent, about 1474. Urbino, Galleria Nazionale delle Marche.
- *David Hilbert*.<sup>s2,p7</sup> This photograph was taken in 1912 for postcards of faculty members at the University of Göttingen which were sold to students at the time (see “Hilbert,” Constance Reid, Springer 1970).

## Chapter 2

- *Apollo 8 Earth picture*.<sup>s2,p6</sup> Earth visible above the lunar surface, taken by Apollo 8 crew member Bill Anders on December 24, 1968.

## Chapter 3

- *Jean Baptiste Joseph Fourier*.<sup>s2,p2</sup> “Portraits et Histoire des Hommes Utiles, Collection de Cinquante Portraits,” Societe Montyon et Franklin, 1839–1840.

- *Josiah Willard Gibbs.*<sup>s2,p2</sup>

**Chapter 4**

- *Claude Shannon.*<sup>s4,p4</sup>

**Chapter 5**

- *Pafnuty Lvovich Chebyshev.*<sup>s2,p1</sup>
- *Pedro, the Voder.*<sup>s5,p8</sup>

**Chapter 6**

- *Werner Karl Heisenberg.*<sup>s2,p2</sup>
- *Dennis Gabor.*<sup>s3,p4</sup>

**Chapter 7**

- *Classic diagram of a dispersion prism.*<sup>s2,p3</sup>

**Chapter 8**

- *MP3 logo.*<sup>s2,p5</sup>
- *WiFi logo.*<sup>s2,p5</sup>

**Chapter 9**

- *The wake after the ferry to Fan, Denmark.*<sup>s2,p3</sup> Credit: Malene Thyssen, <http://commons.wikimedia.org/wiki/User:Malene>.

# Quick Reference

## Abbreviations

AR	Autoregressive
ARMA	Autoregressive moving average
AWGN	Additive white Gaussian noise
BIBO	Bounded input, bounded output
CDF	Cumulative distribution function
DCT	Discrete cosine transform
DFT	Discrete Fourier transform
DTFT	Discrete-time Fourier transform
DWT	Discrete wavelet transform
FFT	Fast Fourier transform
FIR	Finite impulse response
i.i.d.	Independent and identically distributed
IIR	Infinite impulse response
KLT	Karhunen–Loève transform
LOT	Lapped orthogonal transform
LPSV	Linear periodically shift variant
LSI	Linear shift invariant
MA	Moving average
MSE	Mean square error
PDF	Probability density function
POCS	Projection onto convex sets
ROC	Region of convergence
SVD	Singular value decomposition
WSCS	Wide-sense cyclostationary
WSS	Wide-sense stationary

## Abbreviations used in tables and captions but not in the text

FT	Fourier transform
FS	Fourier series
LFT	Local Fourier transform
WT	Wavelet transform

**Elements of Sets**

natural numbers	$\mathbb{N}$	$0, 1, \dots$
integers	$\mathbb{Z}$	$\dots, -1, 0, 1, \dots$
positive integers	$\mathbb{Z}^+$	$1, 2, \dots$
real numbers	$\mathbb{R}$	$(-\infty, \infty)$
positive real numbers	$\mathbb{R}^+$	$(0, \infty)$
complex numbers	$\mathbb{C}$	$a + jb$ or $re^{j\theta}$ with $a, b, r, \theta \in \mathbb{R}$
a generic index set	$\mathcal{I}$	
a generic vector space	$V$	
a generic Hilbert space	$H$	
real part of	$\Re(\cdot)$	
imaginary part of	$\Im(\cdot)$	
closure of set $S$	$\bar{S}$	
functions	$x(t)$	argument $t$ is continuous valued, $t \in \mathbb{R}$
sequences	$x_n$	argument $n$ is an integer, $n \in \mathbb{Z}$
ordered sequence	$(x_n)_n$	
set containing $x_n$	$\{x_n\}_n$	
vector $x$ with $x_n$ as elements	$[x_n]$	
Dirac delta function	$\delta(t)$	$\int_{-\infty}^{\infty} x(t)\delta(t) dt = x(0)$
Kronecker delta sequence	$\delta_n$	$\delta_n = 1$ for $n = 0$ ; $\delta_n = 0$ otherwise
indicator function of interval $I$	$\chi_I(t)$	$\chi_I(t) = 1$ for $t \in I$ ; $\chi_I(t) = 0$ otherwise

**Elements of Real Analysis**

integration by parts	$\int u dv = uv - \int v du$
----------------------	------------------------------

**Elements of Complex Analysis**

complex number	$z$	$a + jb, re^{j\theta}, a, b \in \mathbb{R}, r \in [0, \infty), \theta \in [0, 2\pi)$
conjugation	$z^*$	$a - jb, re^{-j\theta}$
conjugation of coefficients but not of $z$ itself	$X_*(z)$	$X^*(z^*)$
principal root of unity	$W_N$	$e^{-j2\pi/N}$

**Standard Vector Spaces**

Hilbert space of square-summable sequences	$\ell^2(\mathbb{Z})$	$\left\{x : \mathbb{Z} \rightarrow \mathbb{C} \mid \sum_n  x_n ^2 < \infty\right\}$ with inner product $\langle x, y \rangle = \sum_n x_n y_n^*$
Hilbert space of square-integrable functions	$\mathcal{L}^2(\mathbb{R})$	$\left\{x : \mathbb{R} \rightarrow \mathbb{C} \mid \int  x(t) ^2 dt < \infty\right\}$ with inner product $\langle x, y \rangle = \int x(t) y(t)^* dt$
normed vector space of sequences with finite $p$ norm, $1 \leq p < \infty$	$\ell^p(\mathbb{Z})$	$\left\{x : \mathbb{Z} \rightarrow \mathbb{C} \mid \sum_n  x_n ^p < \infty\right\}$ with norm $\ x\ _p = (\sum_n  x_n ^p)^{1/p}$
normed vector space of functions with finite $p$ norm, $1 \leq p < \infty$	$\mathcal{L}^p(\mathbb{R})$	$\left\{x : \mathbb{R} \rightarrow \mathbb{C} \mid \int  x(t) ^p dt < \infty\right\}$ with norm $\ x\ _p = (\int  x(t) ^p dt)^{1/p}$
normed vector space of bounded sequences with supremum norm	$\ell^\infty(\mathbb{Z})$	$\left\{x : \mathbb{Z} \rightarrow \mathbb{C} \mid \sup_n  x_n  < \infty\right\}$ with norm $\ x\ _\infty = \sup_n  x_n $
normed vector space of bounded functions with supremum norm	$\mathcal{L}^\infty(\mathbb{R})$	$\left\{x : \mathbb{R} \rightarrow \mathbb{C} \mid \sup_t  x(t)  < \infty\right\}$ with norm $\ x\ _\infty = \sup_t  x(t) $

**Bases and Frames for Sequences**

standard Euclidean basis	$\{e_n\}$	$e_{n,k} = 1$ , for $k = n$ , and 0 otherwise
vector, element of basis or frame	$\varphi$	when applicable, a <i>column</i> vector
basis or frame	$\Phi$	set of vectors $\{\varphi_n\}$
operator	$\tilde{\Phi}$	concatenation of $\varphi_n$ s in a linear operator: $[\varphi_0 \ \varphi_1 \ \dots \ \varphi_{N-1}]$
vector, element of dual basis or frame	$\tilde{\varphi}$	when applicable, a <i>column</i> vector
	$\tilde{\Phi}$	set of vectors $\{\tilde{\varphi}_n\}$
operator	$\tilde{\Phi}$	concatenation of $\tilde{\varphi}_n$ s in a linear operator: $[\tilde{\varphi}_0 \ \tilde{\varphi}_1 \ \dots \ \tilde{\varphi}_{N-1}]$
expansion in a basis or frame	$x = \Phi \tilde{\Phi}^* x$	

**Transforms**

Fourier transform	$x(t) \xleftrightarrow{\text{FT}} X(\omega)$	$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt$ $x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) e^{j\omega t} d\omega$
Fourier series	$x(t) \xleftrightarrow{\text{FS}} X_k$	$X_k = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-j(2\pi/T)kt} dt$ $x(t) = \sum_{k \in \mathbb{Z}} X_k e^{j(2\pi/T)kt}$
discrete-time Fourier transform	$x_n \xleftrightarrow{\text{DTFT}} X(e^{j\omega})$	$X(e^{j\omega}) = \sum_{n \in \mathbb{Z}} x_n e^{-j\omega n}$ $x_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega$
discrete Fourier transform	$x_n \xleftrightarrow{\text{DFT}} X_k$	$X_k = \sum_{n=0}^{N-1} x_n W_N^{kn}$ $x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k W_N^{-kn}$
local Fourier transform	$x(t) \xleftrightarrow{\text{LFT}} X(\Omega, \tau)$	$X(\Omega, \tau) = \int_{-\infty}^{\infty} x(t) p(t - \tau) e^{-j\Omega t} dt$ $x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} X(\Omega, \tau) g_{\Omega, \tau}(t) d\Omega d\tau$
continuous wavelet transform	$x(t) \xleftrightarrow{\text{CWT}} X(a, b)$	$X(a, b) = \int_{-\infty}^{\infty} x(t) \psi_{a,b}(t) dt$ $x(t) = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^{\infty} X(a, b) \psi_{a,b}(t) \frac{db da}{a^2}$
wavelet series	$x(t) \xleftrightarrow{\text{WS}} \beta_k^{(\ell)}$	$\beta_k^{(\ell)} = \int_{-\infty}^{\infty} x(t) \psi_{\ell,k}(t) dt$ $x(t) = \sum_{\ell \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} \beta_k^{(\ell)} \psi_{\ell,k}(t)$
discrete wavelet transform	$x_n \xleftrightarrow{\text{DWT}} \alpha_k^{(J)}, \beta_k^{(J)}, \dots, \beta_k^{(1)}$	$\alpha_k^{(J)} = \sum_{n \in \mathbb{Z}} x_n g_{n-2^J k}^{(J)}, \beta_k^{(\ell)} = \sum_{n \in \mathbb{Z}} x_n h_{n-2^\ell k}^{(\ell)}$ $x_n = \sum_{k \in \mathbb{Z}} \alpha_k^{(J)} g_{n-2^J k}^{(J)} + \sum_{\ell=1}^J \sum_{k \in \mathbb{Z}} \beta_k^{(\ell)} h_{n-2^\ell k}^{(\ell)}$
discrete cosine transform	$x_n \xleftrightarrow{\text{DCT}} X_k$	$X_0 = \sqrt{\frac{1}{N}} \sum_{n=0}^{N-1} x_n$ $X_k = \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} x_n \cos((2\pi/2N)k(n+1/2))$ $x_0 = \sqrt{\frac{1}{N}} \sum_{k=0}^{N-1} X_k$ $x_n = \sqrt{\frac{2}{N}} \sum_{k=0}^{N-1} X_k \cos((2\pi/2N)k(n+1/2))$
z-transform	$x_n \xleftrightarrow{\text{ZT}} X(z)$	$X(z) = \sum_{n \in \mathbb{Z}} x_n z^{-n}$

**Discrete-Time Nomenclature**

<i>Sequence</i>	$x_n$	signal, vector
<i>Convolution</i>		
linear	$h * x$	$\sum_{k \in \mathbb{Z}} x_k h_{n-k} = \sum_{k \in \mathbb{Z}} h_k x_{n-k}$
circular	$h \circledast x$	$\sum_{k=0}^{N-1} x_k h_{(n-k) \bmod N} = \sum_{k=0}^{N-1} h_k x_{(n-k) \bmod N}$
	$(h * x)_n$	$n$ th element of the convolution result
	$h_{\ell-n} * x_{n-m}$	$\sum_{k \in \mathbb{Z}} x_{k-m} h_{\ell-n+k}$
<i>Eigensequence</i>	$v_n$	eigenfunction, eigenvector
infinite time	$v_n = e^{j\omega n}$	$h * v = H(e^{j\omega}) v$
finite time	$v_n = e^{j2\pi kn/N}$	$h \circledast v = H_k v$
<i>Frequency response</i>		eigenvalue corresponding to $v_n$
infinite time	$H(e^{j\omega})$	$\sum_{n \in \mathbb{Z}} h_n e^{-j\omega n}$
finite time	$H_k$	$\sum_{n=0}^{N-1} h_n e^{-j2\pi kn/N} = \sum_{n=0}^{N-1} h_n W_N^{kn}$

**Continuous-Time Nomenclature**

<i>Function</i>	$x(t)$	signal
<i>Convolution</i>		
linear	$h * x$	$\int_{-\infty}^{\infty} x(\tau) h(t-\tau) d\tau = \int_{-\infty}^{\infty} h(\tau) x(t-\tau) d\tau$
circular	$h \circledast x$	$\int_0^T x(\tau) h(t-\tau) d\tau = \int_0^T h(\tau) x(t-\tau) d\tau$
	$(h * x)(t)$	convolution result at $t$
<i>Eigenfunction</i>	$v(t)$	eigenvector
infinite time	$v(t) = e^{j\omega t}$	$h * v = H(\omega) v$
finite time	$v(t) = e^{j2\pi kt/T}$	$h \circledast v = H_k v$
<i>Frequency response</i>		eigenvalue corresponding to $v(t)$
infinite time	$H(\omega)$	$\int_{-\infty}^{\infty} h(t) e^{-j\omega t} dt$
finite time	$H_k$	$\int_{-T/2}^{T/2} h(\tau) e^{-j2\pi k\tau/T} d\tau$

**Two-Channel Filter Banks***Basic characteristics*

number of channels	$M = 2$
sampling factor	$N = 2$
channel sequences	$\alpha_n \quad \beta_n$

*Filters**Synthesis**Analysis*

	lowpass	highpass	lowpass	highpass
orthogonal	$g_n$	$h_n$	$g_{-n}$	$h_{-n}$
biorthogonal	$g_n$	$h_n$	$\tilde{g}_n$	$\tilde{h}_n$
polyphase components	$g_{0,n}, g_{1,n}$	$h_{0,n}, h_{1,n}$	$\tilde{g}_{0,n}, \tilde{g}_{1,n}$	$\tilde{h}_{0,n}, \tilde{h}_{1,n}$

**Tree-Structured Filter Banks (DWT)***Basic characteristics*

number of channels	$M = J + 1$	
sampling at level $\ell$	$N^{(\ell)} = 2^\ell$	
channel sequences	$\alpha_n^{(J)} \quad \beta_n^{(\ell)}$	$\ell = 1, 2, \dots, J$

*Filters**Synthesis**Analysis*

	lowpass	bandpass <sup>(<math>\ell</math>)</sup>	lowpass	bandpass <sup>(<math>\ell</math>)</sup>	
orthogonal	$g_n^{(J)}$	$h_n^{(\ell)}$	$g_{-n}^{(J)}$	$h_{-n}^{(\ell)}$	
biorthogonal	$g_n^{(J)}$	$h_n^{(\ell)}$	$\tilde{g}_n^{(J)}$	$\tilde{h}_n^{(\ell)}$	
polyphase component $j$	$g_{j,n}^{(J)}$	$h_{j,n}^{(\ell)}$	$\tilde{g}_{j,n}^{(J)}$	$\tilde{h}_{j,n}^{(\ell)}$	$j = 0, 1, \dots, 2^\ell - 1$

**N-Channel Filter Banks***Basic characteristics*

number of channels	$M = N$	
sampling factor	$N$	
channel sequences	$\alpha_{i,n}$	$i = 0, 1, \dots, N - 1$

*Filters**Synthesis**Analysis*

orthogonal filter $i$	$g_{i,n}$	$g_{i,-n}$	
biorthogonal filter $i$	$g_{i,n}$	$\tilde{g}_{i,n}$	
polyphase component $j$	$g_{i,j,n}$	$\tilde{g}_{i,j,n}$	$j = 0, 1, \dots, N - 1$

**Oversampled Filter Banks***Basic characteristics*

number of channels	$M > N$	
sampling factor	$N$	
channel sequences	$\alpha_{i,n}$	$i = 0, 1, \dots, M - 1$

*Filters**Synthesis**Analysis*

filter $i$	$g_{i,n}$	$\tilde{g}_{i,n}$	
polyphase component $j$	$g_{i,j,n}$	$\tilde{g}_{i,j,n}$	$j = 0, 1, \dots, N - 1$



# Preface

The aim of this book is to provide a set of tools for users of state-of-the-art signal processing technology and a solid foundation for those hoping to advance the theory and practice of signal processing. Many of the results and techniques presented here, while rooted in classic Fourier techniques for signal representation, first appeared during a flurry of activity in the 1980s and 1990s. New constructions for local Fourier transforms and orthonormal wavelet bases during that period were motivated both by theoretical interest and by applications, in particular in multimedia communications. New bases with specified time–frequency behavior were found, with impact well beyond the original fields of application. Areas as diverse as computer graphics and numerical analysis embraced some of the new constructions—no surprise given the pervasive role of Fourier analysis in science and engineering.

Now that the dust has settled, some of what was new and esoteric is now fundamental. Our motivation is to bring these new fundamentals to a broader audience to further expand their impact. We thus provide an integrated view of classical Fourier analysis of signals and systems alongside structured representations with time–frequency locality and their myriad of applications.

**Structure of the Book** The book is divided into two parts, the first on foundations and the second on structured representations for signal processing, connected via a bridge—Intermezzo. We have decided to publish the two parts separately, so that the book can be used as soon as possible. While the first part/book can be used independently to cover the foundations of signals and systems, the second part/book relies heavily on the base built in the first part. Thus, these two books are to be seen as integrally related to each other.

**Part I: Foundations of Signals and Systems** This part reviews the necessary mathematical material to make the book self-contained. For many readers, this material might be well known; for others, not, and thus welcome. It is a refresher of the basic mathematical concepts used in signal processing and communications. Thus, in Chapter 1, *From Euclid to Hilbert*, the basic geometric intuition central to Hilbert spaces is reviewed, together with all the necessary tools underlying the construction of bases. Chapter 2, *Sequences and Discrete-Time Systems*, is a crash course on processing signals in discrete time or discrete space. In Chapter 3, *Functions and Continuous-Time Systems*, the mathematics of Fourier transforms and

Fourier series is reviewed. Chapter 4, *Sampling and Interpolation*, talks about the critical link between discrete and continuous domains as given by the sampling theorem and interpolation, while Chapter 5, *Approximation and Compression*, veers from the exact world to the approximate one. The final chapter in Part I, Chapter 6, *Time-Frequency Localization*, considers time-frequency behavior of the abstract representation objects studied thus far

**Intermezzo: Bridging Parts I and II** This short interlude aims to recap the tools seen up to that point, discuss issues arising in the real world as well as ways of adapting these tools for use in the real world. The main concepts seen in Part I, geometry of Hilbert spaces, existence of bases, Fourier representations, sampling and interpolation as well as approximation and compression, build a powerful foundation for modern signal processing. These tools hit roadblocks they must overcome: finiteness and localization, limitations of uncertainty, computational costs.

**Part II: Structured Representations for Signal Processing** This part presents signal representations, including Fourier, local Fourier and wavelet bases, related constructions, as well as frames and continuous transforms.

It starts with Chapter 7, *Filter Banks: Building Blocks of Time-Frequency Expansions*, which presents a thorough treatment of the basic block—the two-channel filter bank, a signal processing device that splits a signal into a coarse, lowpass approximation, and a highpass detail.

We generalize this block in the three chapters that follow, all dealing with Fourier- and wavelet-like representations on sequences: In Chapter 8, *Local Fourier Bases on Sequences*, we discuss Fourier-like bases on sequences, implemented by  $N$ -channel modulated filter banks (first generalization of the two-channel filter banks). In Chapter 9, *Wavelet Bases on Sequences*, we discuss wavelet-like bases on sequences, implemented by tree-structured filter banks (second generalization). In Chapter 10, *Local Fourier and Wavelet Frames on Sequences*, we discuss both Fourier- and wavelet-like frames on sequences, implemented by oversampled filter banks (third generalization).

We then move to the two chapters dealing with Fourier- and wavelet-like representations on functions. In Chapter 11, *Local Fourier Transforms, Frames and Bases on Functions*, we start with the most natural representation of smooth functions with some locality, the local Fourier transform, followed by its sampled version/frame, and leading to results on whether bases are possible. In Chapter 12, *Wavelet Bases, Frames and Transforms on Functions*, we do the same for wavelet representations on functions, but in opposite order: starting from bases, through frames and finally continuous wavelet transform.

The last chapter, Chapter 13, *Approximation, Estimation, and Compression*, uses all the tools we introduced to address state-of-the-art signal processing and communication problems and their solutions. The guiding principle is that there is a domain where the problem at hand will have a sparse solution, at least approximately so. This is known as sparse signal processing, and many examples, from the classical Karhunen-Loève expansion to nonlinear approximation in discrete cosine

transform and wavelet domains, all the way to contemporary research in compressed sensing, use this principle. The chapter introduces and overviews sparse signal processing, covering approximation methods, estimation procedures such as denoising, as well as compression methods and inverse problems.

**Teaching Points** Our aim is to present a synthetic view from basic mathematical principles to actual constructions of bases and frames, always with an eye on concrete applications. While the benefit is a self-contained presentation, the cost is a rather sizable manuscript. To aid with teaching, we provide a reading guide with numerous routes through the material; the levels span from elementary to advanced, but in a gradual fashion and with indications of levels of difficulty. Referencing in the main text is sparse; pointers to bibliography are given in *Further Reading* at the end of each chapter.

The material grew out of teaching signal processing, wavelets and applications in various settings. Two of the authors, Martin Vetterli and Jelena Kovačević, authored a graduate textbook, *Wavelets and Subband Coding*, Prentice Hall, 1995, which they and others used to teach graduate courses at various US and European institutions. This book is online with open access.<sup>1</sup> With more than a decade of experience, the maturing of the field, and the broader interest arising from and for these topics, the time was right for an entirely new text geared towards a broader audience, one that could be used to span levels from undergraduate to graduate, as well as various areas of engineering and science. As a case in point, parts of the text have been used at Carnegie Mellon University in classes on bioimage informatics, where some of the students are life-sciences majors. This plasticity of the text is one of the features which we aimed for, and that most probably differentiates the present book from many others. Another aim is to present side-by-side all methods that arose around signal representations, without favoring any in particular. The truth is that each representation is a tool in the toolbox of the practitioner, and the problem or application at hand ultimately determines the appropriate one to use.

Martin Vetterli, Jelena Kovačević and Vivek K Goyal  
October 2011

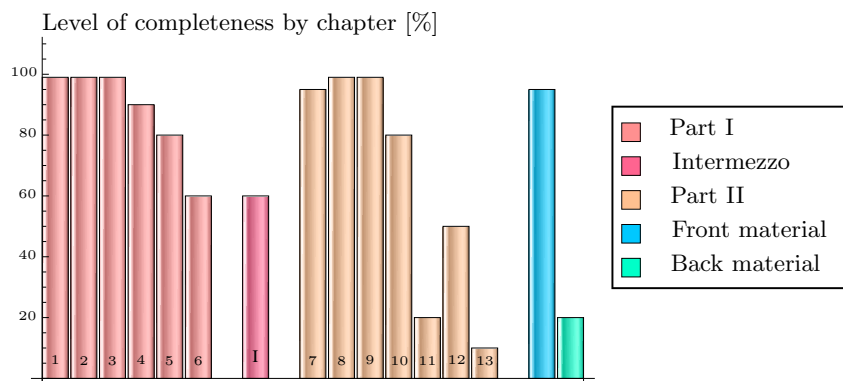
---

<sup>1</sup><http://waveletsandsubbandcoding.org/>



# Release Notes

This is the alpha 3.0 release of the book. Below, we summarize the level of completeness by chapter.



**Front matter** consists of the following items:

- (i) *Image Attribution*  
The section gives credit for all the images not produced by the authors. It is complete, listing all the images in the current version.
- (ii) *Abbreviations*
- (iii) *Quick Reference*  
The section summarizes various concepts used in the book.
- (iv) *Preface*
- (v) *Reading Guide*  
The guide lists several roadmaps of how the book could be used.
- (vi) *From Rainbows to Spectra*  
This chapter is complete.

**Part I** consists of the following chapters:

- (i) *Chapter 1: From Euclid to Hilbert*  
This chapter is complete.
- (ii) *Chapter 2: Sequences and Discrete-Time Systems*  
This chapter is complete.

(iii) *Chapter 3: Functions and Continuous-Time Systems*

This chapter is complete.

(iv) *Chapter 4: Sampling and Interpolation*

This chapter is about 90% complete.

(v) *Chapter 5: Approximation and Compression*

This chapter is about 80% complete.

(vi) *Chapter 6: Time-Frequency Localization*

This chapter is about 60% complete.

**Intermezzo** is a single chapter bridging Parts I and II. It is about 60% complete.

**Part II** consists of the following chapters:

(i) *Chapter 7: Filter Banks: Building Blocks of Time-Frequency Expansions*

This chapter is about 95% complete. Left to finish is Section 7.6 on two-channel filter banks with stochastic inputs.

(ii) *Chapter 8: Local Fourier Bases on Sequences*

This chapter is complete.

(iii) *Chapter 9: Wavelet Bases on Sequences*

This chapter is complete.

(iv) *Chapter 10: Local Fourier and Wavelet Frames on Sequences*

This chapter is about 80% complete. Left to finish are Section 10.6 on computational aspects and a couple of examples.

(v) *Chapter 11: Local Fourier Transform, Frames and Bases on Functions*

This chapter is about 20% complete. Section 11.2 on local Fourier transform is the only one written.

(vi) *Chapter 12: Wavelet Bases, Frames and Transforms on Functions*

This chapter is about 20% complete. Sections 12.1–12.2 are essentially done; Sections 12.3 and 12.5 are written but need major revisions and have notation that does not agree with the rest of the book. Section 12.4 is yet to be written, as are all the components at the end of the chapter (Chapter at a Glance, Historical Remarks and Further Reading).

(vii) *Chapter 13: Approximation, Estimation, and Compression*

An outline of the chapter is included; the chapter is yet to be written.

**Back matter** consists of the following items:

(i) *Bibliography*

The bibliography is complete in the current version.

(ii) *Index*

The index will be generated at the very end.

# Acknowledgments

This project exists thanks to the help of many people, whom we attempt to list below. We apologize for any omissions. The current edition is a work in progress; we welcome corrections, complaints, and suggestions.

We are grateful to Prof. Libero Zuppiroli of EPFL and Christiane Grimm for the photograph that graces the cover; Prof. Zuppiroli proposed an experiment from Newton's treatise on Opticks [106] as emblematic of the book, and Ms. Grimm beautifully photographed the apparatus that he designed. Françoise Behn, Jocelyne Plantefol and Jacqueline Aeberhard typed and organized parts of the manuscript, Eric Strattman assisted with many of the figures, Krista Van Guilder designed and implemented the book web site, and Jorge Albaladejo Pomares design and implemented the book blog. We thank them for their diligence and patience. Patrick Vandewalle designed, wrote and implemented most of the Matlab companion to the book. Similarly, S. Grace Chang and Yann Barbotin helped organize and edit the problem companion. We thank them for their expertise and insight. Whenever possible, we have used Cormac Herley's litmus test for appropriate mathematics in an applied text.

Many instructors have gamely tested pre-alpha versions of this manuscript. Of these, Amina Chebira, Yue M. Lu, and Thao Nguyen have done far more than their share in providing invaluable comments and suggestions. We also thank Zoran Cvetković and Minh Do for teaching with the manuscript and providing many constructive comments, Matthew Fickus for consulting on some finer mathematical points, and Thierry Blu for providing a proof for the Strang-Fix theorem. Useful comments have also been provided by Pedro Aguilar, A. Avudainayagam, Aniruddha Bhargava, S. Esakkirajan, Germán González, Alexandre Haehlen, Mina Karzand, Hossein Rouhani, and Christophe Tournery.

Martin Vetterli thanks EPFL graduate students who helped develop the material, solve problems, catch typos, and suggest improvements, among other things. They include Florence Bénézit, Amina Chebira, Minh Do, Ivan Dokmanic, Pier Luigi Dragotti, Ali Hormati, Ivana Jovanović, Jérôme Lebrun, Pina Marziliano, Fritz Menzer, Reza Parhizkar, Paolo Prandoni, Juri Ranieri, Olivier Roy, Rahul Shukla, Patrick Vandewalle and Vladan Velisavljević. He gratefully acknowledges support from the Swiss National Science Foundation through awards 2000-063664, 200020-103729, 200021-121935, and the European Research Council through award SPARSAM 247006.

Jelena Kovačević thanks her present and past graduate students Ramu Bha-

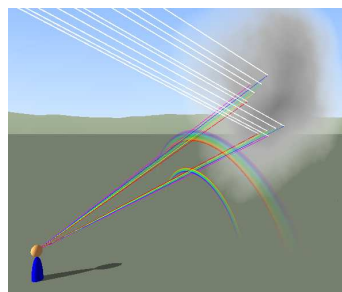
gavatula, Amina Chebira, Jackie Chen, Charles Jackson, Xindian Long, Mike McCann, Anupama Kuruvilla, Tad Merryman, Vivek Oak, Aliaksei Sandryhaila and Gowri Srinivasa, many of whom served as TAs for her classes at CMU, together with Pablo Hennings Yeomans. Thanks also to all the students in the following classes at CMU: 42-431/18-496, 42-731/18-795, 18-799, 42-540, taught from 2003 to 2011. She is grateful to her husband, Giovanni Pacifici, for the many useful scripts automating various book-writing processes. She gratefully acknowledges support from the NSF through awards 1017278, 1130616, 515152, 633775, 0331657, 627250 and 331657; the NIH through awards 1DC010283, EB008870, EB009875 and 1DC010283; the CMU CIT through Infrastructure for Large-Scale Biomedical Computing award; and the PA State Tobacco Settlement, Kamlet-Smith Bioinformatics Grant.

Vivek Goyal thanks TAs and students at MIT for suggestions, in particular Baris Erkmen, Ying-zong Huang, Zahi Karam, Ahmed Kirmani, Ranko Sredojevic, Ramesh Sridharan, Watcharapan Suwansantisuk, Vincent Tan, Archana Venkataraman, Adam Zelinski, and Serhii Zhak. He gratefully acknowledges support from the NSF through awards 0643836 and 0729069; Texas Instruments through its Leadership University Program; MIT through an Esther and Harold E. Edgerton Career Development Chair; and the Centre Bernoulli of EPFL.



# From Rainbows to Spectra

In the 13th century, the Dominican monk, theologian and physicist, Dietrich von Freiberg, performed a simple experiment: he held a spherical bottle filled with water in the sunlight. The bottle played the role of a single water drop, and, following the trajectory of the light that it diffracted and reflected, gave a scientific explanation of the rainbow effect, including the secondary rainbow with weaker reversed colors. He wrote his conclusions in one of his famous treatises, *De iride* (on the rainbow), “probably the most dramatic development of 14th- and 15th-century optics” [63].



Von Freiberg fell short of complete understanding of the rainbow phenomenon because, like many of his contemporaries, he believed that colors were simply intensities between black and white. A full understanding emerged three hundred years later when Descartes and Newton explained that dispersion separates white light into spectral components of different wavelengths—the colors of the rainbow.

This brings us to a central theme of this book: decomposing an entity into its constituent components can be a key step in understanding its essential character. This decomposition can enable even more, such as modifications in the decomposed state. The rainbow’s appearance is explained by the fact that sunlight contains a combination of all wavelengths within the visible range; separation of white light by wavelength, as with a prism, enables modifications prior to recombination. The collection of wavelengths is, as we will see, the spectrum.

A French physicist and mathematician, Joseph Fourier, formalized the notion of the spectrum in the early 19th century. He was interested in the heat equation—the differential equation governing the diffusion of heat. Fourier’s key insight was to decompose a periodic function  $x(t) = x(t + T)$  into an infinite sum of sines and cosines of periods  $T/k$ ,  $k \in \mathbb{Z}^+$ . Since these sine and cosine components are eigenfunctions of the heat equation, the solution of the problem is simplified: one can analyze the differential equation for each component separately and combine the intermediate results, thanks to the linearity of the system. Fourier’s decomposition earned him a coveted prize from the French Academy of Sciences, but with a mention that his work lacked rigor. Indeed, the question of which functions admit a Fourier decomposition is a deep one, and it took many years to settle. Fourier’s work is at

the heart of the present book—for both its strengths and its weaknesses.

**Signal Representations** The idea of a decomposition and a possible modification in the decomposed state leads to signal representations, where signals can be sequences (discrete domain) or functions (continuous domain). Similarly to what Fourier did, where he used sines and cosines for decomposition, we can imagine using other functions with particular properties. Call these basis vectors and denote them by  $\varphi_k$ ,  $k \in \mathbb{Z}$ . Then

$$x = \sum_{k \in \mathbb{Z}} X_k \varphi_k \quad (0.1)$$

is called an expansion of  $x$  in terms of  $\{\varphi_k\}_{k \in \mathbb{Z}}$ .

**Orthonormal Bases** When the basis vectors form an orthonormal set, that is,

$$\langle \varphi_k, \varphi_i \rangle = \delta_{k-i},$$

the coefficients  $X_k$  are obtained from the function  $x$  and the basis vectors  $\varphi_k$  through an inner product

$$X_k = \langle x, \varphi_k \rangle. \quad (0.2)$$

For example, Fourier's construction of a series representation for periodic functions with period  $T = 1$  can be written as

$$x(t) = \sum_{k \in \mathbb{Z}} X_k e^{j2\pi kt}, \quad (0.3a)$$

where

$$X_k = \int_0^1 x(t) e^{-j2\pi kt} dt. \quad (0.3b)$$

We can define basis vectors  $\varphi_k$ ,  $k \in \mathbb{Z}$ , on the interval  $[0, 1)$ , as

$$\varphi_k(t) = e^{j2\pi kt}, \quad 0 \leq t < 1, \quad (0.4)$$

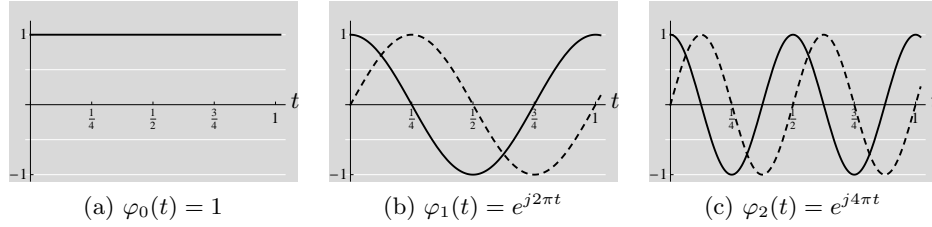
and the Fourier series coefficients as

$$X_k = \langle x, \varphi_k \rangle = \int_0^1 x(t) \varphi_k^*(t) dt = \int_0^1 x(t) e^{-j2\pi kt} dt,$$

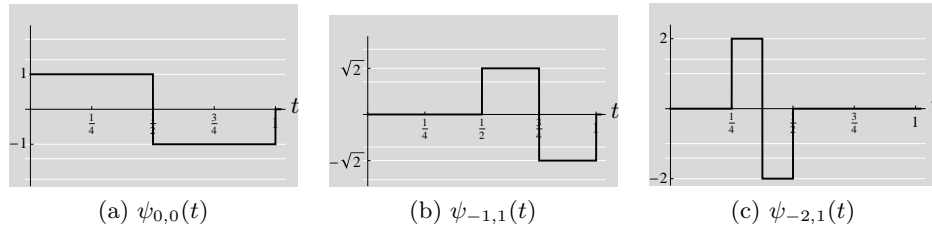
exactly the same as (0.3b). The basis vectors form an orthonormal set (the first few are shown in Figure 0.1):

$$\langle \varphi_k, \varphi_i \rangle = \int_0^1 e^{j2\pi kt} e^{-j2\pi it} dt = \delta_{k-i}. \quad (0.5)$$

While the Fourier series is certainly a key orthonormal basis with many outstanding properties, do other bases exist, and what are their properties? Early in



**Figure 0.1:** The first three Fourier series basis vectors on the interval  $[0, 1)$ . Real parts are shown with solid lines and imaginary parts are shown with dashed lines.



**Figure 0.2:** Example Haar basis functions for the interval  $[0, 1)$ . The prototype function is  $\psi(t) = \psi_{0,0}(t)$ .

the 20th century, Alfred Haar proposed a basis which looks quite different from Fourier's. It is based on a function  $\psi(t)$  defined as

$$\psi(t) = \begin{cases} 1, & \text{for } 0 \leq t < 1/2; \\ -1, & \text{for } 1/2 \leq t < 1; \\ 0, & \text{otherwise.} \end{cases} \quad (0.6)$$

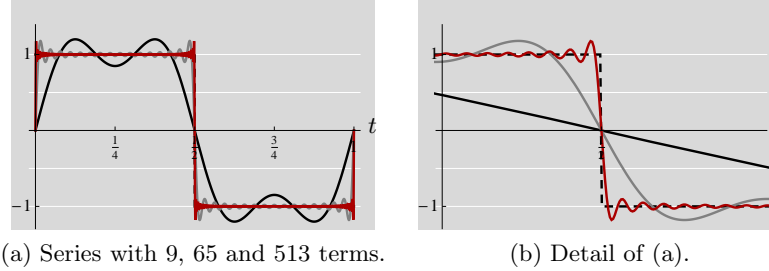
For the interval  $[0, 1)$ , we can build an orthonormal system by scaling  $\psi(t)$  by powers of 2, and then shifting the scaled versions appropriately, yielding

$$\psi_{m,n}(t) = 2^{-m/2} \psi\left(\frac{t - n2^{-m}}{2^{-m}}\right), \quad (0.7)$$

with  $m \in \{0, -1, -2, \dots\}$  and  $n \in \{0, 1, \dots, 2^{-m} - 1\}$  (a few are shown in Figure 0.2). It is quite clear from the figure that the various basis functions are indeed orthogonal to each other, as they either do not overlap, or when they do, one changes sign over the constant span of the other. We will spend a considerable amount of time studying this system in Part II of the book.

While the system (0.7) is certainly orthonormal, it cannot be a basis; for example, on  $[0, 1)$  there would be no way to reconstruct a constant 1. We remedy that by adding the function

$$\varphi_0(t) = \begin{cases} 1, & \text{for } 0 \leq t < 1; \\ 0, & \text{otherwise,} \end{cases}$$



**Figure 0.3:** Gibbs phenomenon for the Fourier series (0.3a) with 9 (black), 65 (gray) and 513 (red) terms of a square wave (Haar basis function  $\psi_{0,0}(t)$  from Figure 0.2(a)).

into the mix, yielding an orthonormal basis for the interval  $[0, 1]$ . This is a very different basis from the Fourier one; for example, instead of being infinitely differentiable, none of the  $\psi_{m,n}$ s is even continuous. We can now define a basis as in (0.3a)-(0.3b)

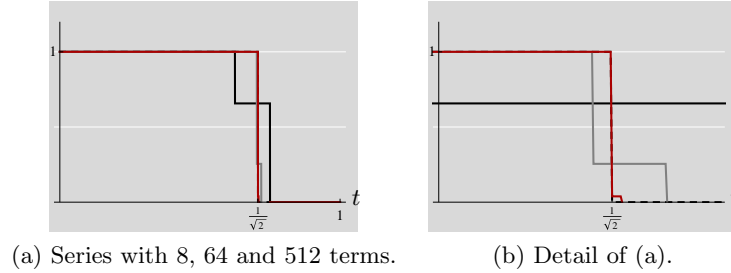
$$x(t) = \langle x, \varphi_0 \rangle \varphi_0(t) + \sum_{m=0}^{-\infty} \sum_{n=0}^{2^m-1} X_{m,n} \psi_{m,n}(t), \quad (0.8a)$$

where

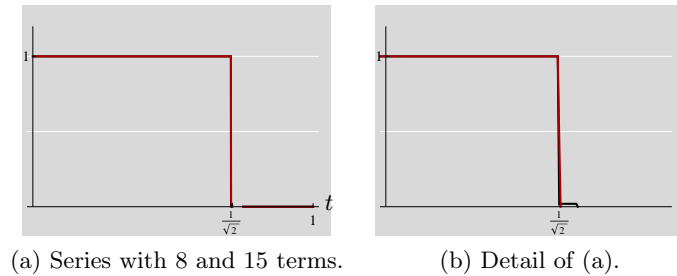
$$X_{m,n} = \int_0^1 x(t) \psi_{m,n}(t) dt. \quad (0.8b)$$

It is natural to ask: Which basis is better? Such a question does not have a simple answer, and the answer will depend on the class of functions or sequences we wish to represent, as well as our goals in the representation. Furthermore, we will have to be careful in describing what we mean by equality in an expansion such as (0.3a); otherwise we could be misled the same way Fourier was.

**Approximation** One way to assess the quality of a basis is to see how well it can approximate a given function with a finite number of terms. History is again enlightening. Fourier series became such a useful tool during the 19th century that researchers built elaborate mechanical devices to compute a function based on Fourier series coefficients. They built analog computers, based on harmonically-related rotating wheels, where amplitudes of Fourier coefficients could be set and the sum computed. One such machine, the Harmonic Integrator, was designed by the physicists Michelson and Stratton, and could compute a series with 80 terms. To the designers' dismay, the synthesis of a square wave from its Fourier series led to oscillations around the discontinuity that would not go away even as they increased the number of terms; they concluded that a mechanical problem was at fault. Not until 1899, when Gibbs proved that Fourier series of discontinuous functions cannot converge uniformly, was this myth dispelled. The phenomenon was termed *Gibbs phenomenon*, referring to the oscillations appearing around the discontinuity when using a finite number of terms (see Figure 0.3 for an example).



**Figure 0.4:** Approximation of a square wave with a Haar basis using the first 8 (black), 64 (gray) and 512 (red) coefficients, with  $m = 0, -1, -2$  and  $n = 0, \dots, 2^{-m} - 1$ . The discontinuity is at the irrational point  $1/\sqrt{2}$ .



**Figure 0.5:** Approximation of a square wave with a Haar basis using 8 (black) and 15 (red) largest-magnitude coefficients (in the neighborhood of the discontinuity). The 15-term approximation is visually indistinguishable from the target function.

So what would the Haar basis provide in this case? Surely, it seems more appropriate for a square wave. Unfortunately, taking the first  $2^{-m}$  coefficients in the natural ordering (the coefficient corresponding to the function  $\varphi(t)$  plus  $2^{-m} - 1$  coefficients corresponding to each scale  $m = 0, -1, -2, \dots$ ) leads to a similarly poor performance, shown in Figure 0.4.

However, changing slightly the approximation procedure makes a big difference. By retaining the largest coefficients in absolute value instead of simply keeping a fixed set of coefficients, the approximation quality changes drastically, as seen in Figure 0.5. Compare Figures 0.3 and 0.4, where the Fourier approximations have 5, 33, and 255 nonzero terms to the similar-quality Haar approximations with only 4, 7, and 10 nonzero terms.

Through this comparison, we have illustrated how the quality of a basis for approximation can depend on the method of approximation. Retaining a predefined set of coefficients, as in the Fourier example case (Figure 0.3) or the first Haar example (Figure 0.4) is called linear approximation. Retaining an adaptive set of coefficients instead, as in the second Haar example (Figure 0.5), is called nonlinear approximation and leads to a superior approximation quality.

The central theme of the book is the design of expansions with certain features. While not the only criterion used to compare expansions, approximation quality arises repeatedly, and we will see that it is closely related to the central signal processing tasks of sampling, filtering, estimation and compression.

# Part I: Foundations of Signals and Systems

The first part of the book defines classes of signals and the basics of representing these signals. This lays the foundation for the second part of the book, in which representations with specific desirable properties are developed and applied.

**Chapter 1: From Euclid to Hilbert**, introduces the basic machinery of Hilbert spaces. These are vector spaces endowed with operations that induce intuitive geometric properties. In this general setting, we develop the notion of signal representations, which are essentially coordinate systems for the vector space. When a representation is complete and not redundant, it provides a *basis* for the space; when it is complete but redundant, it provides a *frame* for the space. A key virtue for a basis is orthonormality; its counterpart for a frame is tightness.

Chapters 2 and 3 narrow our attention to sequence and function spaces that are common in practice while introducing the concept of *time*, leading to an inherent ordering not necessarily present in a general Hilbert space. In **Chapter 2: Sequences and Discrete-Time Systems**, a vector is a sequence that depends on *discrete time*, and an important class of linear operators on these vectors are those that are invariant to time shifts. These operators lead naturally to signal representations using the discrete-time Fourier transform and, for finite-length sequences, the discrete Fourier transform.

**Chapter 3: Functions and Continuous-Time Systems**, parallels Chapter 2: A vector is now a function that depends on *continuous time*, and an important class of linear operators on these vectors are again those that are invariant to time shifts. These operators lead naturally to signal representations using the Fourier transform and, for circularly-extended finite-length functions, or, periodic functions, the Fourier series. The four Fourier representations from these two chapters exemplify the diagonalization of linear, shift-invariant operators, or convolutions, in the various domains.

**Chapter 4: Sampling and Interpolation**, makes fundamental connections between Chapters 2 and 3. Associating a discrete-time sequence to a given continuous-time function is *sampling* and the converse is *interpolation*, central concepts in signal processing as the world is essentially continuous, while digital computations are made on sequences.

**Chapter 5: Approximation and Compression**, introduces many types

of approximations that are central in making computationally-practical tools. Approximation by polynomials and by truncations of series expansions are studied along with the basic principles of compression.

**Chapter 6: Time–Frequency Localization**, introduces time, frequency, scale, and resolution properties of individual vectors, before we construct sets of vectors with which to represent signals in subsequent chapters. These properties build our intuition for what might or might not be possible as properties of representations. In particular, time and frequency localization lead to the concept of a time–frequency plane, where essential differences between Fourier techniques and wavelet techniques become evident: (1) Fourier techniques use vectors with equal spacing in frequency while wavelet techniques do not; and (2) Fourier techniques use vectors at equal scale while wavelet techniques use geometrically-spaced scales.



## Chapter 1

# From Euclid to Hilbert

### Contents

1.1	Introduction . . . . .	10
1.2	Vector Spaces . . . . .	18
1.3	Hilbert Spaces . . . . .	34
1.4	Approximations, Projections, and Decompositions	47
1.5	Bases and Frames . . . . .	65
1.6	Computational Aspects . . . . .	113
1.A	Elements of Analysis and Topology . . . . .	128
1.B	Elements of Linear Algebra . . . . .	133
1.C	Elements of Probability . . . . .	144
	Chapter at a Glance . . . . .	151
	Historical Remarks . . . . .	152
	Further Reading . . . . .	152
	Exercises with Solutions . . . . .	153
	Exercises . . . . .	157

We start our journey into the world of Fourier and wavelets with different backgrounds and perspectives. This chapter aims to establish a common language, develop the foundations for our study, and begin to draw out key themes.

There will be more formal definitions in this chapter than in any other, to approach the ideal of a self-contained treatment. However, we must assume some background in common: On the one hand, we expect the reader to be familiar with linear algebra at the level of [140, Ch. 1–5] and probability at the level of [11, Ch. 1–4]. (The textbooks we have cited are just examples; nothing unique to those books is necessary.) On the other hand, we are not assuming prior knowledge of general vector space abstractions or mathematical analysis beyond basic calculus; we develop these topics here to extend geometric intuition from ordinary Euclidean space to spaces of sequences and functions. For more details on abstract vector spaces, we recommend books by Kreyszig [93], Luenberger [98], and Young [176].

## 1.1 Introduction

This section introduces many topics of the chapter through the familiar setting of the real plane. In the more general treatment of subsequent sections, the intuition we have developed through years of dealing with the Euclidean spaces around us ( $\mathbb{R}^2$  and  $\mathbb{R}^3$ ), will generalize to some not-so-familiar spaces. Readers comfortable with vector spaces, inner products, norms, projections, and bases, may skip this section; otherwise, this will be a gentle introduction into Euclid's world.

### Real Plane as a Vector Space

Let us start with a look at the familiar setting of  $\mathbb{R}^2$ , that is, real vectors with two coordinates. We adopt the convention of vectors being columns and often write them compactly as transposes of rows, such as  $x = [x_0 \ x_1]^T$ . The first entry is the horizontal component and the second entry is the vertical component.

Adding two vectors in the plane produces a third one also in the plane; multiplying a vector by a real scalar produces a second vector also in the plane. These two ingrained facts make the real plane be a *vector space*.

### Inner Product and Norm

The *inner product* of vectors  $x = [x_0 \ x_1]^T$  and  $y = [y_0 \ y_1]^T$  in the real plane is

$$\langle x, y \rangle = x_0 y_0 + x_1 y_1. \quad (1.1)$$

Other names for inner product are *scalar product* and *dot product*. The inner product of a vector with itself is simply

$$\langle x, x \rangle = x_0^2 + x_1^2,$$

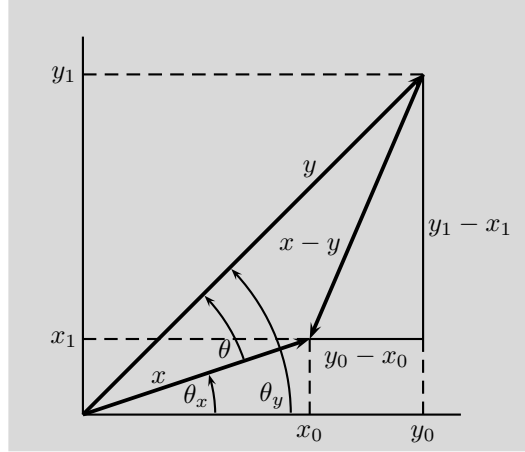
a nonnegative quantity that is zero when  $x_0 = x_1 = 0$ . The *norm* of a vector  $x$  is

$$\|x\| = \sqrt{\langle x, x \rangle}. \quad (1.2)$$

While the norm is sometimes called the *length*, we avoid this usage because length can also refer to the number of components in a vector. A vector with norm 1 is called a *unit vector*.

In (1.1), the inner product computation depends on the choice of coordinate axes. Let us now derive an expression in which the coordinates disappear. Consider  $x$  and  $y$  as shown in Figure 1.1. Define the angle between  $x$  and the positive horizontal axis as  $\theta_x$  (measured counterclockwise), and define  $\theta_y$  similarly. Using a little algebra and trigonometry, we get

$$\begin{aligned} \langle x, y \rangle &= x_0 y_0 + x_1 y_1 \\ &= (\|x\| \cos \theta_x)(\|y\| \cos \theta_y) + (\|x\| \sin \theta_x)(\|y\| \sin \theta_y) \\ &= \|x\| \|y\| (\cos \theta_x \cos \theta_y + \sin \theta_x \sin \theta_y) \\ &= \|x\| \|y\| \cos(\theta_x - \theta_y). \end{aligned} \quad (1.3)$$



**Figure 1.1:** A pair of vectors in  $\mathbb{R}^2$ .

Thus, the inner product of the two vectors is the product of their norms and the cosine of the angle  $\theta = \theta_x - \theta_y$  between them.

The inner product measures both the norms of the vectors and the similarity of their orientations. For fixed vector norms, the greater the inner product, the closer the vectors are in orientation. The orientations are closest when the vectors are colinear and pointing in the same direction, that is, when  $\theta = 0$ ; they are the farthest when the vectors are antiparallel, that is, when  $\theta = \pi$ . When  $\langle x, y \rangle = 0$ , the vectors are called *orthogonal* or *perpendicular*. From (1.3), we see that  $\langle x, y \rangle$  is zero only when the norm of one vector is zero (meaning one of the vectors is the vector  $[0 \ 0]^T$ ) or the cosine of the angle between them is zero ( $\theta = \pm\pi/2$ ). So at least in the latter case, this is consistent with the conventional concept of perpendicularity.

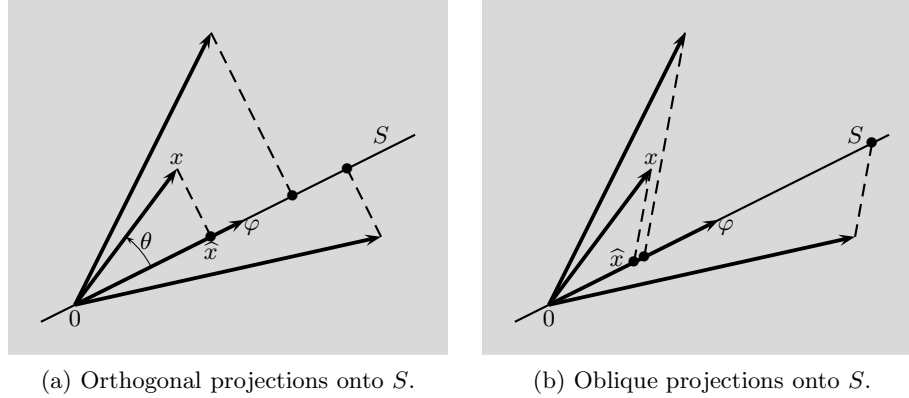
The *distance* between two vectors is defined as the norm of their difference:

$$d(x, y) = \|x - y\| = \sqrt{\langle x - y, x - y \rangle} = \sqrt{(x_0 - y_0)^2 + (x_1 - y_1)^2}. \quad (1.4)$$

### Subspaces and Projections

A line through the origin is the simplest case of a *subspace*, and *projection* to a subspace is intimately related to inner products.

Starting with a vector  $x$  and applying an *orthogonal projection operator* onto some subspace results in the vector  $\hat{x}$  closest to  $x$  (among all vectors in the subspace). The connection to orthogonality is that the difference between the vector and its orthogonal projection  $x - \hat{x}$  is orthogonal to every vector in the subspace. Orthogonal projection is illustrated in Figure 1.2(a). The subspace  $S$  is formed by the scalar multiples of the vector  $\varphi$ , and three orthogonal projections onto  $S$  are shown. As depicted, the action of the operator is like looking at the shadow that



**Figure 1.2:** Examples of projections onto the subspace  $S$  specified by the unit vector  $\varphi$ .

the input vector casts on  $S$  when light rays are orthogonal to  $S$ . This operation is linear, meaning that the orthogonal projection of  $x + y$  equals the sum of the orthogonal projections of  $x$  and  $y$ . Also, the orthogonal projection operator leaves vectors in  $S$  unchanged.

Given a unit vector  $\varphi$ , the *orthogonal projection* onto the subspace specified by  $\varphi$  is  $\hat{x} = \langle x, \varphi \rangle \varphi$ . This can also be written as

$$\hat{x} \stackrel{(a)}{=} \langle x, \varphi \rangle \varphi = (\|x\| \|\varphi\| \cos \theta) \varphi = (\|x\| \cos \theta) \varphi, \quad (1.5)$$

where (a) uses  $\|\varphi\| = 1$ , and  $\theta$  is the angle measured counterclockwise from  $\varphi$  to  $x$ , as marked in Figure 1.2(a). When  $\varphi$  is not of unit norm, the orthogonal projection onto the subspace specified by  $\varphi$  is

$$\hat{x} \stackrel{(a)}{=} (\|x\| \cos \theta) \frac{\varphi}{\|\varphi\|} = (\|x\| \|\varphi\| \cos \theta) \frac{\varphi}{\|\varphi\|^2} \stackrel{(b)}{=} \frac{1}{\|\varphi\|^2} \langle x, \varphi \rangle \varphi, \quad (1.6)$$

where (a) expresses the orthogonal projection using the unit vector  $(\varphi/\|\varphi\|)$ , and (b) uses (1.3).

Projection is more general than orthogonal projection; for example, Figure 1.2(b) illustrates *oblique projection*. The operator is still linear and vectors in the subspace are still left unchanged; however, the difference  $(x - \hat{x})$  is not orthogonal to  $S$  anymore.

### Bases and Coordinates

We defined the real plane as a vector space using coordinates: the first coordinate is the signed distance as measured from left to right, and the second coordinate is the signed distance as measured from bottom to top. In doing so, we implicitly used the *standard basis*  $e_0 = [1 \ 0]^T$ ,  $e_1 = [0 \ 1]^T$ , which is a particular orthonormal

basis of  $\mathbb{R}^2$ . Expressing vectors in a variety of bases is central to our study, and vectors' coordinates will differ depending on the choice of basis.

**Orthonormal Bases** Vectors  $e_0 = [1 \ 0]^T$  and  $e_1 = [0 \ 1]^T$  constituting the standard basis are depicted in Figure 1.3(a). They are orthogonal and of unit norm and are thus called *orthonormal*. We have been using this basis implicitly in that

$$x = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = x_0 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + x_1 \begin{bmatrix} 0 \\ 1 \end{bmatrix} = x_0 e_0 + x_1 e_1 \quad (1.7)$$

is an *expansion* of  $x$  with respect to the basis  $\{e_0, e_1\}$ . For this basis, it is obvious that an expansion exists for any  $x$  because the *coefficients* of the expansion  $x_0$  and  $x_1$  are simply “read off” of  $x$ .

The general condition for  $\{\varphi_0, \varphi_1\}$  to be an orthonormal basis is

$$\langle \varphi_i, \varphi_k \rangle = \delta_{i-k} \quad \text{for } i, k \in \{0, 1\}, \quad (1.8)$$

where  $\delta_{i-k}$  is a convenient shorthand defined as<sup>2</sup>

$$\delta_{i-k} = \begin{cases} 1, & \text{for } i = k; \\ 0, & \text{otherwise.} \end{cases} \quad (1.9)$$

From the  $i \neq k$  cases, the basis vectors are orthogonal to each other; from the  $i = k$  cases, they are of unit norm. With any orthonormal basis  $\{\varphi_0, \varphi_1\}$ , one can uniquely find the coefficients of the expansion

$$x = \alpha_0 \varphi_0 + \alpha_1 \varphi_1$$

simply through the inner products

$$\alpha_0 = \langle x, \varphi_0 \rangle \quad \text{and} \quad \alpha_1 = \langle x, \varphi_1 \rangle.$$

The resulting coefficients satisfy

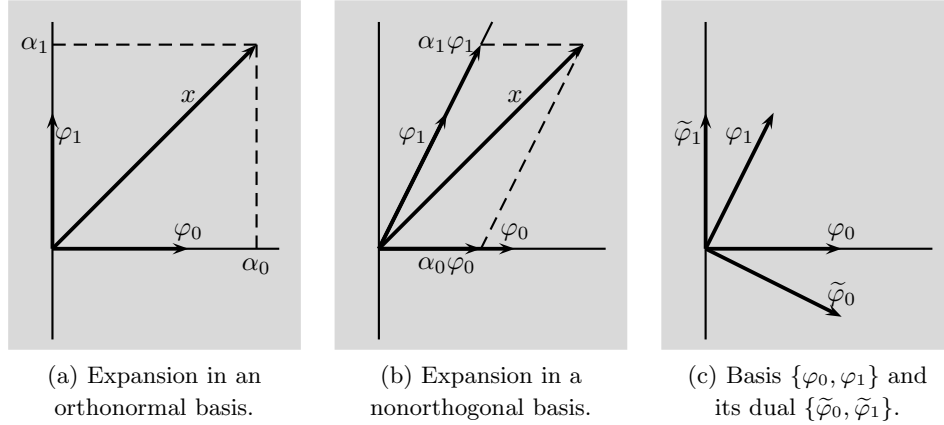
$$|\alpha_0|^2 + |\alpha_1|^2 = \|x\|^2 \quad (1.10)$$

by the Pythagorean theorem, because  $\alpha_0$  and  $\alpha_1$  form sides of a right triangle with hypotenuse of length  $\|x\|$  (Figure 1.3(a)). The equality (1.10) is an example of Parseval's equality<sup>3</sup> and is related to Bessel's inequality; these will be formally introduced in Section 1.5.2.

**Biorthogonal Pairs of Bases** Expansions like (1.7) do not necessarily need  $\{\varphi_0, \varphi_1\}$  to be orthonormal. As an example, consider the problem of representing an arbitrary vector  $x = [x_0 \ x_1]^T$  as an expansion  $\alpha_0 \varphi_0 + \alpha_1 \varphi_1$  with respect to  $\varphi_0 = [1 \ 0]^T$  and  $\varphi_1 = [\frac{1}{2} \ 1]^T$  (see Figure 1.3(b)). This is not a trivial exercise such

<sup>2</sup> $\delta_n$  is called the *Kronecker delta* sequence and is formally defined in Chapter 2, (2.7).

<sup>3</sup>What we call Parseval's equality in this book is sometimes called Plancherel's equality as well.

**Figure 1.3:** Expansions in  $\mathbb{R}^2$ .

as the one of expanding with the standard basis, but we can still come up with an intuitive procedure.

Since  $\varphi_0$  has no vertical component, we should use  $\varphi_1$  to match the vertical component of  $x$ , yielding  $\alpha_1 = x_1$ . (This is illustrated with the diagonal dashed line in Figure 1.3(b).) Then, we need  $\alpha_0 = x_0 - \frac{1}{2}x_1$  for the horizontal component to be correct. We can express what we have just done with inner products as

$$\alpha_0 = \langle x, \tilde{\varphi}_0 \rangle \quad \text{and} \quad \alpha_1 = \langle x, \tilde{\varphi}_1 \rangle,$$

where

$$\tilde{\varphi}_0 = \begin{bmatrix} 1 & -\frac{1}{2} \end{bmatrix}^T \quad \text{and} \quad \tilde{\varphi}_1 = \begin{bmatrix} 0 & 1 \end{bmatrix}^T,$$

with vectors  $\tilde{\varphi}_0$  and  $\tilde{\varphi}_1$  as shown in Figure 1.3(c). We have thus just derived an instance of the expansion formula

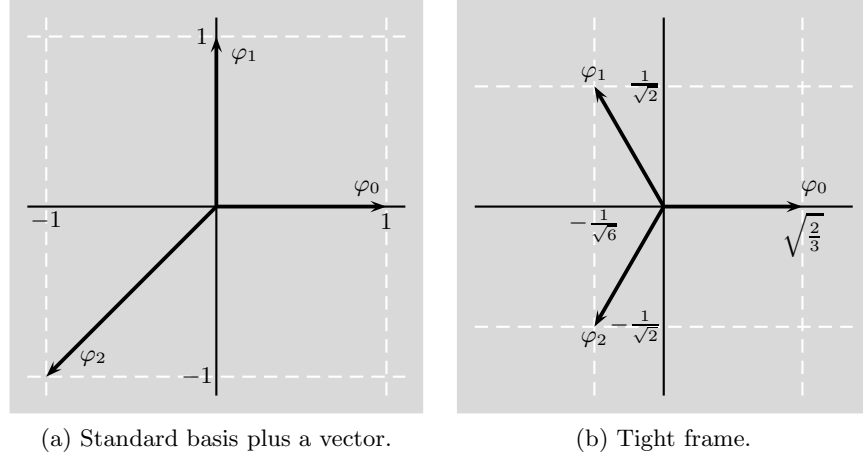
$$x = \alpha_0 \varphi_0 + \alpha_1 \varphi_1 = \langle x, \tilde{\varphi}_0 \rangle \varphi_0 + \langle x, \tilde{\varphi}_1 \rangle \varphi_1, \quad (1.11)$$

where  $\{\tilde{\varphi}_0, \tilde{\varphi}_1\}$  is the basis *dual* to the basis  $\{\varphi_0, \varphi_1\}$ , and the two bases form a *biorthogonal pair of bases*. For any basis, the dual basis is unique. The defining characteristic for a biorthogonal pair is

$$\langle \tilde{\varphi}_i, \varphi_k \rangle = \delta_{i-k}. \quad (1.12)$$

You can check that this is satisfied in our example and that any orthonormal basis is its own dual. Clearly, designing a biorthogonal basis pair has more degrees of freedom than designing an orthonormal basis. The disadvantage is that (1.10) does not hold, and furthermore, computations can become numerically unstable if  $\varphi_0$  and  $\varphi_1$  are too close to colinear.

An expansion like (1.11) is often termed a *change of basis*, since it expresses  $x$  with respect to  $\{\varphi_0, \varphi_1\}$ , rather than in the standard basis  $\{e_0, e_1\}$ . In other words, the coefficients  $(\alpha_0, \alpha_1)$  are the coordinates of  $x$  in this new basis  $\{\varphi_0, \varphi_1\}$ .



**Figure 1.4:** Illustrations of overcomplete sets of vectors (frames).

**Frames** The signal expansion (1.11) has the minimum possible number of terms to work for every  $x \in \mathbb{R}^2$ —two terms because the dimension of the space is two. It can also be useful to have an expansion of the form

$$x = \langle x, \tilde{\varphi}_0 \rangle \varphi_0 + \langle x, \tilde{\varphi}_1 \rangle \varphi_1 + \langle x, \tilde{\varphi}_2 \rangle \varphi_2. \quad (1.13)$$

Here, an expansion will exist as long as  $\{\varphi_0, \varphi_1, \varphi_2\}$  are not colinear. Then, even after the set  $\{\varphi_0, \varphi_1, \varphi_2\}$  is fixed, there are infinitely many dual sets  $\{\tilde{\varphi}_0, \tilde{\varphi}_1, \tilde{\varphi}_2\}$  such that (1.13) holds for all  $x \in \mathbb{R}^2$ . Such redundant sets are called *frames* and their (nonunique) dual sets *dual frames*. This flexibility can be used in various ways. For example, setting a component of  $\tilde{\varphi}_i$  to zero could save a multiplication and an addition in computing an expansion, or the dual, which we said was not unique, could be chosen to make the coefficients as small as possible.

As an example, let us start with the standard basis  $\{\varphi_0 = e_0, \varphi_1 = e_1\}$ , add a vector  $\varphi_2 = -e_0 - e_1$  to it, and see what happens (see Figure 1.4(a)):

$$\varphi_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \varphi_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \varphi_2 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}. \quad (1.14)$$

As there are now three vectors in  $\mathbb{R}^2$ , they are linearly dependent; indeed,  $\varphi_2 = -\varphi_0 - \varphi_1$ . Moreover, these three vectors must be able to represent every vector in  $\mathbb{R}^2$  since each two-element subset is able to do so. To show that, we use the expansion  $x = \langle x, \varphi_0 \rangle \varphi_0 + \langle x, \varphi_1 \rangle \varphi_1$  and add a zero to it to read:

$$x = \langle x, \varphi_0 \rangle \varphi_0 + \langle x, \varphi_1 \rangle \varphi_1 + \underbrace{(\langle x, \varphi_1 \rangle - \langle x, \varphi_1 \rangle) \varphi_0 + (\langle x, \varphi_1 \rangle - \langle x, \varphi_1 \rangle) \varphi_1}_{=0}.$$

We now rearrange it slightly:

$$x = \langle x, (\varphi_0 + \varphi_1) \rangle \varphi_0 + \langle x, 2\varphi_1 \rangle \varphi_1 + \langle x, \varphi_1 \rangle (-\varphi_0 - \varphi_1) = \sum_{k=0}^2 \langle x, \tilde{\varphi}_k \rangle \varphi_k,$$

with  $\tilde{\varphi}_0 = \varphi_0 + \varphi_1$ ,  $\tilde{\varphi}_1 = 2\varphi_1$ ,  $\tilde{\varphi}_2 = -\varphi_0 - \varphi_1$ . This expansion is exactly of the form (1.13) and is reminiscent of the one for biorthogonal pairs of bases, which we have seen earlier, except that the vectors involved in the expansion are now linearly dependent. This shows that indeed, we can expand any  $x \in \mathbb{R}^2$  in terms of the frame  $\{\varphi_0, \varphi_1, \varphi_2\}$  and one of its possible dual frames  $\{\tilde{\varphi}_0, \tilde{\varphi}_1, \tilde{\varphi}_2\}$ .

Can we now get a frame to somehow mimic an orthonormal basis? Consider:

$$\varphi_0 = \begin{bmatrix} \sqrt{\frac{2}{3}} \\ 0 \end{bmatrix}, \quad \varphi_1 = \begin{bmatrix} -\frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}, \quad \varphi_2 = \begin{bmatrix} -\frac{1}{\sqrt{6}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix}, \quad (1.15)$$

shown in Figure 1.4(b). By expanding an arbitrary  $x = [x_0 \ x_1]^T$ , we can verify that  $x = \sum_{k=0}^2 \langle x, \varphi_k \rangle \varphi_k$  holds for any  $x$ . The expansion looks like the orthonormal basis one, where the same set of vectors plays both roles (inside the inner product and outside). The norm is preserved similarly to what happens with orthonormal bases ( $\sum_{k=0}^2 |\langle x, \varphi_k \rangle|^2 = \|x\|^2$ ), except that the norms of the frame vectors are not 1, but rather  $\sqrt{2/3}$ . A frame with this property is called a *tight frame*. We could have renormalized the frame vectors by  $\sqrt{3/2}$  to make them unit-norm vectors, in which case  $\sum_{k=0}^2 |\langle x, \varphi_k \rangle|^2 = (3/2)\|x\|^2$ , where  $3/2$  indicates the redundancy of the frame (we have  $3/2$  times more vectors than needed for an expansion in  $\mathbb{R}^2$ ).

**Matrix View of Bases and Frames** An expansion with a basis or frame involves operations that can be expressed conveniently with matrices.

Take the biorthogonal basis expansion formula (1.11). The coefficients in the expansion are the inner products

$$\begin{aligned} \alpha_0 &= \langle x, \tilde{\varphi}_0 \rangle = \tilde{\varphi}_{00} x_0 + \tilde{\varphi}_{01} x_1 \\ \alpha_1 &= \langle x, \tilde{\varphi}_1 \rangle = \tilde{\varphi}_{10} x_0 + \tilde{\varphi}_{11} x_1, \end{aligned}$$

where  $\tilde{\varphi}_0 = [\tilde{\varphi}_{00} \ \tilde{\varphi}_{01}]^T$  and  $\tilde{\varphi}_1 = [\tilde{\varphi}_{10} \ \tilde{\varphi}_{11}]^T$ . Rewrite the above as a matrix-vector product

$$\begin{bmatrix} \alpha_0 \\ \alpha_1 \end{bmatrix} = \begin{bmatrix} \langle x, \tilde{\varphi}_0 \rangle \\ \langle x, \tilde{\varphi}_1 \rangle \end{bmatrix} = \underbrace{\begin{bmatrix} \tilde{\varphi}_{00} & \tilde{\varphi}_{01} \\ \tilde{\varphi}_{10} & \tilde{\varphi}_{11} \end{bmatrix}}_{\tilde{\Phi}^T} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}.$$

The matrix  $\tilde{\Phi}^T$  with  $\tilde{\varphi}_0^T$  and  $\tilde{\varphi}_1^T$  as rows is called the *analysis operator*, and left multiplying by it computes the expansion coefficients  $(\alpha_0, \alpha_1)$  in the basis  $\{\varphi_0, \varphi_1\}$ .

The reconstruction of  $x$  from  $(\alpha_0, \alpha_1)$  is through

$$x = \alpha_0 \varphi_0 + \alpha_1 \varphi_1.$$



This can be written with a matrix–vector product as

$$\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \alpha_0 \begin{bmatrix} \varphi_{00} \\ \varphi_{01} \end{bmatrix} + \alpha_1 \begin{bmatrix} \varphi_{10} \\ \varphi_{11} \end{bmatrix} = \underbrace{\begin{bmatrix} \varphi_{00} & \varphi_{10} \\ \varphi_{01} & \varphi_{11} \end{bmatrix}}_{\Phi} \begin{bmatrix} \alpha_0 \\ \alpha_1 \end{bmatrix},$$

where  $\varphi_0 = [\varphi_{00} \ \varphi_{01}]^T$  and  $\varphi_1 = [\varphi_{10} \ \varphi_{11}]^T$ . The matrix  $\Phi$  with  $\varphi_0$  and  $\varphi_1$  as columns is called the *synthesis operator*, and left multiplying by it performs the reconstruction of  $x$  from  $(\alpha_0, \alpha_1)$ .

The matrix view makes it obvious that the expansion formula (1.11) holds for any  $x \in \mathbb{R}^2$  when  $\Phi\tilde{\Phi}^T$  is the identity matrix. In other words, we must have  $\Phi^{-1} = \tilde{\Phi}^T$ , which is equivalent to (1.12). The inverse exists whenever  $\{\varphi_0, \varphi_1\}$  is a basis, and inverting  $\Phi$  determines the dual basis  $\{\tilde{\varphi}_0, \tilde{\varphi}_1\}$ .

In the case of an orthonormal basis,  $\Phi^{-1} = \Phi^T$ , that is, the matrix–vector equations above hold with  $\tilde{\Phi} = \Phi$ .

The case of a 3-element frame is similar, with matrices  $\Phi$  and  $\tilde{\Phi}$  each having 2 rows and 3 columns. The validity of the expansion (1.13) hinges on  $\Phi$  being a left inverse of  $\tilde{\Phi}^T$ . In the example we saw earlier,

$$\Phi = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix}, \quad (1.16a)$$

and its dual frame was

$$\tilde{\Phi} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}. \quad (1.16b)$$

Such a left inverse,  $\tilde{\Phi}^T$ , is never unique; thus dual frames are not unique. For example, the following dual frame

$$\tilde{\Phi} = \begin{bmatrix} 0 & -1 & -1 \\ -1 & 0 & -1 \end{bmatrix}, \quad (1.16c)$$

would work as well.

## Chapter Outline

The next several sections follow the progression of topics in this brief introduction: In Section 1.2, we formally introduce vector spaces and equip them with inner products and norms. We also give several examples of common vector spaces. In Section 1.3, we discuss the concept of completeness that turns an inner product space into a Hilbert space. More importantly, we define the central concept of orthogonality and then introduce linear operators. We follow with approximations, projections and decompositions in Section 1.4. In Section 1.5, we define bases and frames. This step gives us the tools to analyze signals and to create approximate representations. Section 1.5.5 develops the matrix view of basis and frame expansions. Section 1.6 discusses a few algorithms pertaining to the material covered. The appendices review some elements of analysis and topology, linear algebra, as well as probability.

## 1.2 Vector Spaces

Sets of mathematical objects can be highly abstract, and imposing the axioms of a normed vector space is amongst the simplest ways to induce useful structure. Furthermore, we will see that images, audio signals, and many other types of signals can be modeled and manipulated well using vector space models. This section introduces vector spaces formally, including inner products, norms, and metrics. We give pointers to reference texts in *Further Reading*.

### 1.2.1 Definition and Properties

A vector space is any set with objects, called vectors, that can be added and scaled while staying within the set. The formal definition of a vector space needs to specify the field of scalars and properties that are required of the vector addition and scalar multiplication operations.

**DEFINITION 1.1 (VECTOR SPACE)** A vector space over a field of scalars  $\mathbb{C}$  (or  $\mathbb{R}$ ) is a set of vectors,  $V$ , together with operations of vector addition and scalar multiplication. For any  $x, y, z$  in  $V$  and  $\alpha, \beta$  in  $\mathbb{C}$  (or  $\mathbb{R}$ ), these operations must satisfy the following properties:

- (i) *Commutativity*:  $x + y = y + x$ .
- (ii) *Associativity*:  $(x + y) + z = x + (y + z)$  and  $(\alpha\beta)x = \alpha(\beta x)$ .
- (iii) *Distributivity*:  $\alpha(x + y) = \alpha x + \alpha y$  and  $(\alpha + \beta)x = \alpha x + \beta x$ .

Furthermore, the following hold:

- (iv) *Additive identity*: There exists an element  $\mathbf{0}$  in  $V$ , such that  $x + \mathbf{0} = \mathbf{0} + x = x$ , for every  $x$  in  $V$ .
- (v) *Additive inverse*: For every  $x$  in  $V$ , there exists a unique element  $-x$  in  $V$ , such that  $x + (-x) = (-x) + x = \mathbf{0}$ .
- (vi) *Multiplicative identity*: For every  $x$  in  $V$ ,  $1 \cdot x = x$ .

We have used the bold  $\mathbf{0}$  to emphasize that the zero vector is different than the zero scalar. In later chapters we will drop this distinction. The definition of a vector space requires the field of scalars to be specified; we opted to carry real and complex numbers in parallel. This will be true for a number of other definitions in this chapter as well. We now discuss some common vector spaces.

#### $\mathbb{C}^N$ : Vector Space of Complex-Valued Finite-Dimensional Vectors

$$\mathbb{C}^N = \left\{ x = [x_0 \ x_1 \ \dots \ x_{N-1}]^T \mid x_n \in \mathbb{C}, n \in \{0, 1, \dots, N-1\} \right\}, \quad (1.17a)$$

## 1.2. Vector Spaces

19

where the vector addition and scalar multiplication are defined componentwise,

$$\begin{aligned} x + y &= [x_0 \ x_1 \ \dots \ x_{N-1}]^T + [y_0 \ y_1 \ \dots \ y_{N-1}]^T \\ &= [x_0 + y_0 \ x_1 + y_1 \ \dots \ x_{N-1} + y_{N-1}]^T, \\ \alpha x &= \alpha [x_0 \ x_1 \ \dots \ x_{N-1}]^T = [\alpha x_0 \ \alpha x_1 \ \dots \ \alpha x_{N-1}]^T. \end{aligned}$$

It is easy to verify that the six properties in Definition 1.1 hold;  $\mathbb{C}^N$  is thus a vector space (see also Solved Exercise 1.1). The definition of the standard Euclidean space,  $\mathbb{R}^N$ , follows similarly, except over  $\mathbb{R}$ .

 **$\mathbb{C}^{\mathbb{Z}}$ : Vector Space of Complex-Valued Sequences over  $\mathbb{Z}$** 

$$\mathbb{C}^{\mathbb{Z}} = \left\{ x = [\dots \ x_{-1} \ \boxed{x_0} \ x_1 \ \dots]^T \mid x_n \in \mathbb{C}, n \in \mathbb{Z} \right\}, \quad (1.17b)$$

where the vector addition and scalar multiplication are defined componentwise.<sup>4</sup>

 **$\mathbb{C}^{\mathbb{R}}$ : Vector Space of Complex-Valued Functions over  $\mathbb{R}$** 

$$\mathbb{C}^{\mathbb{R}} = \left\{ x(t) \mid x(t) \in \mathbb{C}, t \in \mathbb{R} \right\}, \quad (1.17c)$$

with the natural addition and scalar multiplication operations:

$$(x + y)(t) = x(t) + y(t), \quad (1.18a)$$

$$(\alpha x)(t) = \alpha x(t). \quad (1.18b)$$

Other vector spaces of sequences and functions can be denoted similarly, for example,  $\mathbb{C}^{\mathbb{N}}$  for complex-valued sequences indexed from 0,  $\mathbb{C}^{\mathbb{R}^+}$  for complex-valued functions on the positive real line,  $\mathbb{C}^{[a,b]}$  for complex-valued functions on the interval  $[a, b]$ , etc.

The operations of vector addition and scalar multiplication seen above can be used to define many other vector spaces. For example, componentwise addition and multiplication can be used to define the vector space of matrices, while the natural operations of additions and scalar multiplication of functions can be used to define the vector space of polynomials:

**EXAMPLE 1.1** Fix a positive integer  $N$  and consider the real-valued polynomials of degree at most  $(N - 1)$ ,  $x(t) = \sum_{k=0}^{N-1} \alpha_k t^k$ . These form a vector space over  $\mathbb{R}$  under the natural addition and multiplication operations. Since each polynomial is specified by its coefficients, polynomials combine exactly like vectors in  $\mathbb{R}^N$ .

<sup>4</sup>When writing infinite sequences as column vectors, the entry with index zero is boxed to serve as a reference point. We will do this also for infinite matrices.

**DEFINITION 1.2 (SUBSPACE)** A subset  $S$  of a vector space  $V$  is a subspace when it is closed under the operations of vector addition and scalar multiplication:

- (i) For all  $x$  and  $y$  in  $S$ ,  $x + y$  is in  $S$ .
- (ii) For all  $x$  in  $S$  and  $\alpha$  in  $\mathbb{C}$  (or  $\mathbb{R}$ ),  $\alpha x$  is in  $S$ .

A subspace  $S$  is itself a vector space over the same field of scalars as  $V$  and with the same vector addition and scalar multiplication operations as  $V$ .

**EXAMPLE 1.2 (SUBSPACES)**

- (i) Let  $x$  be a vector in a vector space  $V$ . The set of vectors of the form  $\alpha x$  with  $\alpha \in \mathbb{C}$  is a subspace.
- (ii) In the vector space of complex-valued sequences over  $\mathbb{Z}$ , the sequences that are zero outside of  $\{2, 3, 4, 5\}$  form a subspace. The same can be said with  $\{2, 3, 4, 5\}$  replaced by any finite or infinite subset of the domain  $\mathbb{Z}$ .
- (iii) In the vector space of real-valued functions on  $\mathbb{R}$ , the functions that are constant on intervals  $[k - \frac{1}{2}, k + \frac{1}{2})$ ,  $k \in \mathbb{Z}$ , form a subspace. This is because the sum of two functions each of which is constant on  $[k - \frac{1}{2}, k + \frac{1}{2})$  is also a function constant on  $[k - \frac{1}{2}, k + \frac{1}{2})$ , while a function constant on  $[k - \frac{1}{2}, k + \frac{1}{2})$  multiplied by a scalar is also a function constant on  $[k - \frac{1}{2}, k + \frac{1}{2})$ .
- (iv) In the vector space of real-valued functions on the interval  $[-\frac{1}{2}, \frac{1}{2}]$  under the natural operations of addition and scalar multiplication (1.18), the set of odd functions,

$$S_{\text{odd}} = \{x(t) \mid x(t) = -x(-t) \text{ for all } t \in [-\frac{1}{2}, \frac{1}{2}]\}, \quad (1.19a)$$

is a subspace. Similarly, the set of even functions,

$$S_{\text{even}} = \{x(t) \mid x(t) = x(-t) \text{ for all } t \in [-\frac{1}{2}, \frac{1}{2}]\}, \quad (1.19b)$$

is also a subspace. Either is easily checked as the sum of two odd (even) functions yields an odd (even) function; scalar multiplication of an odd (even) function yields again an odd (even) function.

**DEFINITION 1.3 (AFFINE SUBSPACE)** A subset  $T$  of a vector space  $V$  is an affine subspace when there exist a vector  $x \in V$  and a subspace  $S \subset V$  such that any  $t \in T$  can be written as  $x + s$  for some  $s \in S$ .

Beware that an affine subspace is not necessarily a subspace; it is a subspace if and only if it includes  $\mathbf{0}$ . Affine subspaces generalize the concept of a plane in Euclidean geometry; subspaces correspond just to planes that include the origin. Affine subspaces are *convex sets*, meaning that if vectors  $x$  and  $y$  are in the set, so is any vector  $\theta x + (1 - \theta)y$  for  $\theta \in [0, 1]$ .

## 1.2. Vector Spaces

21

## EXAMPLE 1.3 (AFFINE SUBSPACES)

- (i) Let  $x$  and  $y$  be vectors in a vector space  $V$ . The set of vectors of the form  $x + \alpha y$  with  $\alpha \in \mathbb{C}$  is an affine subspace.
- (ii) In the vector space of complex-valued sequences over  $\mathbb{Z}$ , the sequences that equal 1 outside of  $\{2, 3, 4, 5\}$  form an affine subspace.

The definition of a subspace is suggestive of one way in which subspaces arise—by combining a finite number of vectors in  $V$ . A set of all finite linear combinations of elements in that set is a *span*, which we now define.

DEFINITION 1.4 (SPAN) The span of a set of vectors  $S$  is the set of all finite linear combinations of vectors in  $S$ :

$$\text{span}(S) = \left\{ \sum_{k=0}^{N-1} \alpha_k \varphi_k \mid \alpha_k \in \mathbb{C} \text{ (or } \mathbb{R}), \varphi_k \in S \text{ and } N \in \mathbb{N} \right\}.$$

Note that a span is always a subspace and that the sum has a finite number of terms even if the set  $S$  is infinite.

EXAMPLE 1.4 Proper subspaces (those smaller than the entire space) arise in linear algebra when one looks at matrix–vector products with rank-deficient matrices. Consider the vector space  $V = \mathbb{R}^N$  and let  $S = \{y = Ax \mid x \in \mathbb{R}^N\}$ . Here,  $A$  are size- $(N \times N)$  real-valued matrices  $A$  of rank  $M$ , with  $M < N$ . Applying the conditions in the definition of a subspace (Definition 1.2) to  $S$ , and using the properties of matrix multiplication, we verify that  $S$  is indeed an  $M$ -dimensional subspace of  $\mathbb{R}^N$ . As per (1.206a) in Appendix 1.B, this subspace is the span of the columns of  $A$ .

Many different sets can have the same span, and it can be of fundamental interest to find the smallest set with a particular span. This leads to the dimension of a vector space, which depends on the concept of linear independence.

DEFINITION 1.5 (LINEAR INDEPENDENCE) The set of vectors  $\{\varphi_0, \varphi_1, \dots, \varphi_{N-1}\}$  is called linearly independent when  $\sum_{k=0}^{N-1} \alpha_k \varphi_k = 0$  is true only if  $\alpha_k = 0$  for all  $k$ . Otherwise, the set is linearly dependent. An infinite set of vectors is called linearly independent when every finite subset is linearly independent.

DEFINITION 1.6 (DIMENSION) A vector space  $V$  is said to have dimension  $N$  when it contains a linearly independent set with  $N$  elements and every set with  $N+1$  or more elements is linearly dependent. If no such finite  $N$  exists, the vector space is infinite dimensional.

### 1.2.2 Inner Product

Our intuition from Euclidean spaces goes farther than just adding and multiplying. It has geometric notions of orientation and orthogonality as well as metric notions of norm and distance. In this and the next subsection, we extend these to our abstract spaces.

As visualized in Figure 1.2, an inner product is like a signed norm of an orthogonal projection of one vector onto a subspace spanned by another. It thus measures norm along with relative orientation.

**DEFINITION 1.7 (INNER PRODUCT)** An inner product on a vector space  $V$  over  $\mathbb{C}$  (or  $\mathbb{R}$ ) is a complex-valued (or real-valued) function  $\langle \cdot, \cdot \rangle$  defined on  $V \times V$  with the following properties for any  $x, y, z \in V$  and  $\alpha \in \mathbb{C}$  (or  $\mathbb{R}$ ):

- (i) *Distributivity*:  $\langle x + y, z \rangle = \langle x, z \rangle + \langle y, z \rangle$ .
- (ii) *Linearity in the first argument*:  $\langle \alpha x, y \rangle = \alpha \langle x, y \rangle$ .
- (iii) *Hermitian symmetry*:  $\langle x, y \rangle^* = \langle y, x \rangle$ .
- (iv) *Positive definiteness*:  $\langle x, x \rangle \geq 0$ , and  $\langle x, x \rangle = 0$  if and only if  $x = \mathbf{0}$ .

Note that (ii) and (iii) imply  $\langle x, \alpha y \rangle = \alpha^* \langle x, y \rangle$ . Thus, along with being linear in the first argument, the inner product is conjugate-linear in the second argument.<sup>5</sup> Also note that the inner product being a number excludes the possibility of it being a divergent (infinite) quantity. Thus, an inner product on  $V$  must return a finite number for every pair of vectors in  $V$ . This constrains both the functional form of an inner product as well as the set of vectors to which it can be applied.

**EXAMPLE 1.5 (INNER PRODUCT)** Consider the vector space  $\mathbb{C}^2$ .

- (i)  $\langle x, y \rangle = x_0 y_0^* + 5x_1 y_1^*$  is a valid inner product; it satisfies all the conditions of Definition 1.7.
- (ii)  $\langle x, y \rangle = x_0^* y_0 + x_1^* y_1$  is not a valid inner product; it violates Definition 1.7(ii). For example, if  $x = y = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$  and  $\alpha = j$ ,  $\langle \alpha x, y \rangle = -j$  and  $\alpha \langle x, y \rangle = j$ , and thus,  $\langle \alpha x, y \rangle \neq \alpha \langle x, y \rangle$ .
- (iii)  $\langle x, y \rangle = x_0 y_0^*$  is not a valid inner product; it violates Definition 1.7(iv) because  $x = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$  is nonzero yet yields  $\langle x, x \rangle = 0$ .

**Standard Inner Product on  $\mathbb{C}^N$**  The standard inner product on  $\mathbb{C}^N$  is

$$\langle x, y \rangle = \sum_{n=0}^{N-1} x_n y_n^* = y^* x, \quad (1.20a)$$

<sup>5</sup>We have used the convention that is dominant in mathematics; in physics, on the other hand, the inner product is defined to be linear in the second argument and conjugate-linear in the first.

where the second equality uses matrix–vector multiplication to express the sum, with vectors implicitly column vectors and  $*$  denoting the Hermitian transpose operation. While we will use this inner product frequently and without special mention, this is not the only valid inner product for  $\mathbb{C}^N$  (or  $\mathbb{R}^N$ ) (see Exercise 1.6).

**Standard Inner Product on  $\mathbb{C}^{\mathbb{Z}}$**  The standard inner product on the vector space of complex-valued sequences over  $\mathbb{Z}$  is

$$\langle x, y \rangle = \sum_{n \in \mathbb{Z}} x_n y_n^* = y^* x, \quad (1.20b)$$

where we are taking the unusual step of using matrix product notation with an infinite row vector  $y^*$  and an infinite column vector  $x$ . As stated above, the sum must converge to a finite number for the inner product to be valid, restricting the set of vectors on which we can operate.

**Standard Inner Product on  $\mathbb{C}^{\mathbb{R}}$**  The standard inner product on the vector space of complex-valued functions over  $\mathbb{R}$  is

$$\langle x, y \rangle = \int_{-\infty}^{\infty} x(t) y^*(t) dt. \quad (1.20c)$$

We must be careful that the integral exists and is finite for the inner product to be valid, restricting the set of vectors on which we can operate. We restrict this set even further to those functions with a countable number of discontinuities, thus eliminating a number of subtle technical issues.<sup>6</sup>

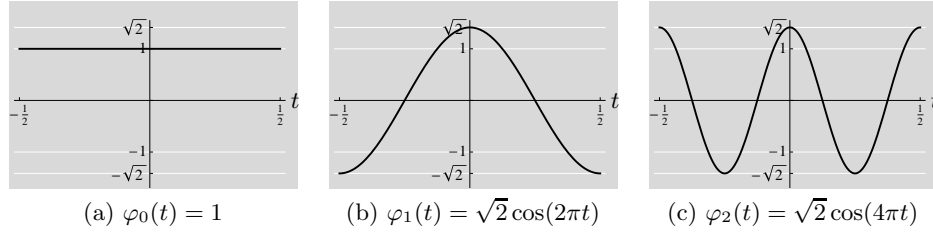
### Orthogonality

An inner product endows a space with the geometric properties of orientation, for example perpendicularity and angles. In particular, an inner product being zero has special significance.

#### DEFINITION 1.8 (ORTHOGONALITY)

- (i) Vectors  $x$  and  $y$  are said to be orthogonal when  $\langle x, y \rangle = 0$ , written as  $x \perp y$ .
- (ii) A set of vectors  $S$  is called orthogonal when  $x \perp y$  for every  $x$  and  $y$  in  $S$  such that  $x \neq y$ .
- (iii) A set of vectors  $S$  is called orthonormal when it is orthogonal and  $\langle x, x \rangle = 1$  for every  $x$  in  $S$ .

<sup>6</sup>For the inner product to be positive definite, as per Definition 1.7(iv), we must identify any function satisfying  $\int_{-\infty}^{\infty} |x(t)|^2 dt = 0$  with 0. From this point on, we restrict our attention to Lebesgue-measurable functions, and all integrals should be seen as Lebesgue integrals. In other words, we exclude from consideration those functions that are not “well behaved” in the above sense. This restriction is not unduly stringent for any practical purpose. We follow the creed of R. W. Hamming [66]: “...if whether an airplane would fly or not depended on whether some function ... was Lebesgue but not Riemann integrable, then I would not fly in it.”

**Figure 1.5:** Example functions from (1.21).

- (iv) A vector  $x$  is said to be orthogonal to a set of vectors  $S$  when  $x \perp s$  for all  $s \in S$ , written as  $x \perp S$ .
- (v) Two sets  $S_0$  and  $S_1$  are said to be orthogonal when every vector  $s_0$  in  $S_0$  is orthogonal to the set  $S_1$ , written as  $S_0 \perp S_1$ .
- (vi) Given a subspace  $S$  of a vector space  $V$ , the orthogonal complement of  $S$ , denoted  $S^\perp$ , is the set  $\{x \in V \mid x \perp S\}$ .

Note that the set  $S^\perp$  is a subspace as well. Vectors in an orthonormal set  $\{\varphi_k\}_{k \in \mathbb{Z}}$  are linearly independent, since  $\mathbf{0} = \sum \alpha_k \varphi_k$  implies that  $0 = \langle \sum \alpha_k \varphi_k, \varphi_k \rangle = \sum \alpha_k \langle \varphi_k, \varphi_k \rangle = \alpha_k$  for any  $k$ .

**EXAMPLE 1.6 (ORTHOGONALITY)** Consider the set of vectors  $\Phi = \{\varphi_k\}_{k \in \mathbb{N}}$ , where

$$\varphi_0(t) = 1, \quad (1.21a)$$

$$\varphi_k(t) = \sqrt{2} \cos(2\pi kt), \quad k = 1, 2, \dots \quad (1.21b)$$

The functions  $\varphi_0$ ,  $\varphi_1$ , and  $\varphi_2$  are shown in Figure 1.5. Using inner product (1.20c), we have the following properties:

- (i) For any  $k, m \in \mathbb{Z}^+$  with  $k \neq m$ , vectors  $\varphi_k$  and  $\varphi_m$  are orthogonal because

$$\begin{aligned}
 \langle \varphi_k, \varphi_m \rangle &= 2 \int_{-1/2}^{1/2} \cos(2\pi kt) \cos(2\pi mt) dt \\
 &\stackrel{(a)}{=} \int_{-1/2}^{1/2} [\cos(2\pi(k+m)t) + \cos(2\pi(k-m)t)] dt \\
 &= \frac{1}{2\pi} \left[ \frac{1}{k+m} \sin(2\pi(k+m)t) \Big|_{-1/2}^{1/2} + \frac{1}{k-m} \sin(2\pi(k-m)t) \Big|_{-1/2}^{1/2} \right] \\
 &= 0,
 \end{aligned}$$

where (a) follows from the trigonometric identity for the product of cosines. It is easy to check that for any  $k \in \mathbb{Z}^+$ , the vectors  $\varphi_0$  and  $\varphi_k$  are orthogonal.



- (ii) The set of vectors  $\Phi$  is orthogonal because, using (i),  $\varphi_k \perp \varphi_m$  for every  $\varphi_k$  and  $\varphi_m$  in  $\Phi$  such that  $\varphi_k \neq \varphi_m$ .
- (iii) The set of vectors  $\Phi$  is orthonormal because it is orthogonal, as we have just shown in (ii), and for any  $k = 1, 2, \dots$ :

$$\begin{aligned} \langle \varphi_k, \varphi_k \rangle &= 2 \int_{-1/2}^{1/2} (\cos(2\pi kt))^2 dt \stackrel{(a)}{=} 2 \int_{-1/2}^{1/2} \frac{1 + \cos(4\pi kt)}{2} dt \\ &= \left[ t \Big|_{-1/2}^{1/2} + \frac{1}{4\pi k} \sin(4\pi kt) \Big|_{-1/2}^{1/2} \right] = 1, \end{aligned}$$

where (a) follows from the double-angle formula for cosine. The inner product of  $\varphi_0$  with itself is trivially 1.

- (iv) For any  $k \in \mathbb{N}$ , the vector  $\varphi_k$  is orthogonal to the set of odd functions  $S_{\text{odd}}$  defined in (1.19a) because  $\varphi_k$  is orthogonal to every  $s \in S_{\text{odd}}$ :

$$\begin{aligned} \langle \varphi_k, s \rangle &= \int_{-1/2}^{1/2} \sqrt{2} \cos(2\pi kt) s(t) dt \\ &= \sqrt{2} \left( \int_{-1/2}^0 \cos(2\pi kt) s(t) dt + \int_0^{1/2} \cos(2\pi kt) s(t) dt \right) \\ &\stackrel{(a)}{=} \sqrt{2} \left( - \int_{-1/2}^0 \cos(2\pi kt) s(-t) dt + \int_0^{1/2} \cos(2\pi kt) s(t) dt \right) \\ &\stackrel{(b)}{=} \sqrt{2} \left( - \int_0^{1/2} \cos(2\pi k\tau) s(\tau) d\tau + \int_0^{1/2} \cos(2\pi kt) s(t) dt \right) = 0, \end{aligned}$$

where (a) follows from the definition of an odd function; and (b) from the change of variable  $\tau = -t$  and the fact that cosine is an even function, that is,  $\cos(-2\pi k\tau) = \cos(2\pi k\tau)$ .

- (v) The set  $\Phi$  is orthogonal to the set of odd functions  $S_{\text{odd}}$  because each vector in  $\Phi$  is orthogonal to  $S_{\text{odd}}$ , using (iv).

### Inner Product Spaces

A vector space equipped with an inner product from Definition 1.7 becomes an *inner product space* (sometimes also called a *pre-Hilbert space*). As we have mentioned, on  $\mathbb{C}^{\mathbb{Z}}$  and  $\mathbb{C}^{\mathbb{R}}$ , we must exercise caution and choose the subspace for which the inner product is finite.

### 1.2.3 Norm

A norm is a function that assigns a length, or size, to a vector (analogously to the magnitude of a scalar).

**DEFINITION 1.9 (NORM)** A norm on a vector space  $V$  over  $\mathbb{C}$  (or  $\mathbb{R}$ ) is a real-valued function  $\|\cdot\|$  defined on  $V$  with the following properties for any  $x, y \in V$  and  $\alpha \in \mathbb{C}$  (or  $\mathbb{R}$ ):

- (i) *Positive definiteness*:  $\|x\| \geq 0$ , and  $\|x\| = 0$  if and only if  $x = \mathbf{0}$ .
- (ii) *Positive scalability*:  $\|\alpha x\| = |\alpha| \|x\|$ .
- (iii) *Triangle inequality*:  $\|x + y\| \leq \|x\| + \|y\|$ , with equality if and only if  $y = \alpha x$ .

Note that the same comments we made about the finiteness and validity of an inner product apply to the norm as well.

In the above, the triangle inequality got its name because it has the following geometric interpretation: the length of any side of a triangle is smaller than or equal to the sum of the lengths of the other two sides; equality occurs only when two sides are colinear, that is, when the triangle degenerates into a line segment. For example, if  $V = \mathbb{C}$  and the standard norm is used, the triangle inequality becomes:

$$|x + y| \leq |x| + |y| \quad \text{for any } x, y \in \mathbb{C}. \quad (1.22)$$

An inner product can be used to define a norm, which we say is the norm *induced* by the inner product. The three inner products we have seen in (1.20) induce corresponding standard norms in  $\mathbb{C}^N$ ,  $\mathbb{C}^{\mathbb{Z}}$  and  $\mathbb{C}^{\mathbb{R}}$ , respectively. Not all norms are induced by inner products. We will see examples of this both here as well as in Section 1.2.4.

**EXAMPLE 1.7 (NORM)** Consider the vector space  $\mathbb{C}^2$ .

- (i)  $\|x\| = |x_0|^2 + 5|x_1|^2$  is a valid norm; it satisfies all the conditions of Definition 1.9. It is induced by the inner product from Example 1.5(i).
- (ii)  $\|x\| = |x_0| + |x_1|$  is a valid norm. However, it is not induced by any inner product.
- (iii)  $\|x\| = |x_0|$  is not a valid norm; it violates Definition 1.9(i) because  $x = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$  is nonzero yet yields  $\|x\| = 0$ .

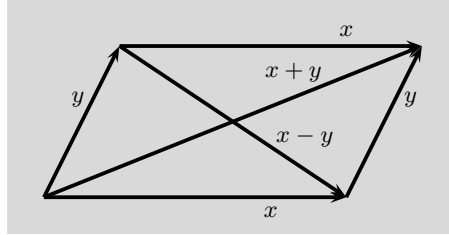
**Standard Norm on  $\mathbb{C}^N$**  The standard norm on  $\mathbb{C}^N$ , induced by the inner product (1.20a), is:

$$\|x\| = \sqrt{\langle x, x \rangle} = \left( \sum_{n=0}^{N-1} |x_n|^2 \right)^{1/2}. \quad (1.23a)$$

This is also called the *Euclidean norm* and yields the conventional notion of length.

**Standard Norm on  $\mathbb{C}^{\mathbb{Z}}$**  The standard norm on  $\mathbb{C}^{\mathbb{Z}}$ , induced by the inner product (1.20b), is:

$$\|x\| = \sqrt{\langle x, x \rangle} = \left( \sum_{n \in \mathbb{Z}} |x_n|^2 \right)^{1/2}. \quad (1.23b)$$



**Figure 1.6:** Illustration of the parallelogram law.

**Standard Norm on  $\mathbb{C}^{\mathbb{R}}$**  The standard norm on  $\mathbb{C}^{\mathbb{R}}$ , induced by the inner product (1.20c), is:

$$\|x\| = \sqrt{\langle x, x \rangle} = \left( \int_{-\infty}^{\infty} |x(t)|^2 dt \right)^{1/2}. \quad (1.23c)$$

### Properties of Norms Induced by an Inner Product

The following facts hold in any inner product space.

**Cauchy–Schwarz Inequality** This widely used inequality states that:<sup>7</sup>

$$|\langle x, y \rangle| \leq \|x\| \|y\|, \quad (1.24)$$

with equality if and only if  $x = \alpha y$  for some scalar  $\alpha$ . We will see an example of the Cauchy–Schwarz inequality shortly.

**Parallelogram Law** This law generalizes from Euclidean geometry to an inner product space and states that (see Figure 1.6 for an illustration):

$$\|x + y\|^2 + \|x - y\|^2 = 2(\|x\|^2 + \|y\|^2). \quad (1.25)$$

Be careful that even though no inner product appears in the parallelogram law, it necessarily holds only in an inner product space. In fact, (1.25) is a necessary and sufficient condition for a norm to be induced by an inner product (see Exercise 1.10).

**Pythagorean Theorem** Like the parallelogram law, this theorem generalizes from Euclidean geometry to an inner product space. The statement learned in elementary school involves the sides of a right triangle. In its more general form the theorem

<sup>7</sup>The result for sums is due to Cauchy, while the result for integrals is due to Schwarz. Buniakovsky seems to have published the result for integrals six years earlier than Schwarz; thus, the integral version is sometimes referred to as the *Buniakovsky inequality*.

states that:<sup>8</sup>

$$x \perp y \quad \text{implies} \quad \|x + y\|^2 = \|x\|^2 + \|y\|^2. \quad (1.26a)$$

Among many possible proofs of the theorem, one follows from expanding  $\langle x + y, x + y \rangle$  into four terms and noting that  $\langle x, y \rangle = \langle y, x \rangle = 0$  because of orthogonality. By induction, the Pythagorean theorem holds in a more general form for any countable set of orthogonal vectors:

$$\{x_k\}_{k \in \mathcal{K}} \text{ orthogonal} \quad \text{implies} \quad \left\| \sum_{k \in \mathcal{K}} x_k \right\|^2 = \sum_{k \in \mathcal{K}} \|x_k\|^2. \quad (1.26b)$$

### Normed Vector Spaces

A vector space equipped with a norm becomes a *normed vector space*. As with the inner product, we must exercise caution and choose the subspace for which the norm is finite.

### Metric

Intuitively, the length of a vector can be thought of as the vector's distance from the origin. This extends naturally to a metric induced by a norm, or a distance.

**DEFINITION 1.10 (METRIC, OR DISTANCE)** In a normed vector space, the metric, or distance between vectors  $x$  and  $y$  is the norm of their difference:

$$d(x, y) = \|x - y\|.$$

Much as norms induced by inner products are a small fraction of all possible norms, metrics induced by norms are a small fraction of all possible metrics. In this book, we will have no need for more general concepts of metric; for the interested reader, Exercise 1.13 gives the axioms that a metric must satisfy and explores metrics that are not induced by norms.

### 1.2.4 Standard Spaces

We now discuss some standard vector spaces: first inner product spaces (they are also normed vector spaces as their inner products induce the corresponding norms), followed by other normed vector spaces (those for which the norms are not induced by an inner product).

<sup>8</sup>The theorem was found on a Babylonian tablet circa 1900–1600 B.C., and it is not clear whether Pythagoras himself or one of his disciples stated and later proved the theorem. The first written proof and reference to the theorem are in Euclid's *Elements* [68].

**Standard Inner Product Spaces**

The first three spaces,  $\mathbb{C}^N$ ,  $\ell^2(\mathbb{Z})$  and  $\mathcal{L}^2(\mathbb{R})$ , are the spaces most often used in this book.<sup>9</sup> For each, the inner product and norm have been defined already in (1.20) and (1.23); we repeat them for each space for easy reference.

**$\mathbb{C}^N$ : Space of Complex-Valued Finite-Dimensional Vectors** We repeat the expressions for the inner product (1.20a) and the norm (1.23a) it induces:

$$\langle x, y \rangle = \sum_{n=0}^{N-1} x_n y_n^*, \quad \|x\| = \left( \sum_{n=0}^{N-1} |x_n|^2 \right)^{1/2}. \quad (1.27)$$

The above norm is not the only norm possible on  $\mathbb{C}^N$ ; in the next subsection, we will introduce  $p$  norms as possible alternatives.

**$\ell^2(\mathbb{Z})$ : Space of Square-Summable Sequences** This is the normed vector space of square-summable complex-valued sequences, and it uses the inner product (1.20b) and the norm (1.23b):

$$\langle x, y \rangle = \sum_{n \in \mathbb{Z}} x_n y_n^*, \quad \|x\| = \left( \sum_{n \in \mathbb{Z}} |x_n|^2 \right)^{1/2}. \quad (1.28)$$

This space is often referred to as the space of *finite-energy sequences*.

By the Cauchy–Schwarz inequality (1.24), the finiteness of  $\|x\|$  and  $\|y\|$  for any  $x$  and  $y$  in  $\ell^2(\mathbb{Z})$  implies the inner product  $\langle x, y \rangle$  is finite, provided the sum in the inner product is well defined. A somewhat technical point is that the square-summability condition that determines which sequences are in  $\ell^2(\mathbb{Z})$  also ensures that the sum in the inner product is indeed well defined; see Exercise 1.12.

**$\mathcal{L}^2(\mathbb{R})$ : Space of Square-Integrable Functions** This is the normed vector space of square-integrable complex-valued functions, and it uses the inner product (1.20c) and the norm (1.23c):

$$\langle x, y \rangle = \int_{-\infty}^{\infty} x(t) y^*(t) dt, \quad \|x\| = \left( \int_{-\infty}^{\infty} |x(t)|^2 dt \right)^{1/2}. \quad (1.29)$$

This space is often referred to as the space of *finite-energy functions*. According to Definition 1.6, this space is infinite dimensional; for example,  $\{e^{-t^2}, te^{-t^2}, t^2e^{-t^2}, \dots\}$  are linearly independent. As in the case of  $\ell^2(\mathbb{Z})$ , the inner product is always well defined.

We can restrict the domain  $\mathbb{R}$  to just an interval  $[a, b]$ , in which case the space becomes  $\mathcal{L}^2([a, b])$ , that is, the space of complex-valued square-integrable functions

<sup>9</sup>The reasoning behind naming  $\ell^2(\mathbb{Z})$  and  $\mathcal{L}^2(\mathbb{R})$  will become clear in the section on standard normed vector spaces shortly.

on the interval  $[a, b]$ . The inner product and norm follow naturally from (1.29):

$$\langle x, y \rangle = \int_a^b x(t)y^*(t) dt, \quad \|x\| = \left( \int_a^b |x(t)|^2 dt \right)^{1/2}. \quad (1.30)$$

In  $\mathcal{L}^2([a, b])$ , the Cauchy–Schwarz inequality (1.24) becomes

$$\left| \int_a^b x(t)y^*(t) dt \right| \leq \left( \int_a^b |x(t)|^2 dt \right)^{1/2} \left( \int_a^b |y(t)|^2 dt \right)^{1/2}, \quad (1.31)$$

with equality if and only if  $x(t)$  and  $y(t)$  are linearly dependent. By setting  $y(t) = 1$  and squaring both sides, we get another useful fact:

$$\left| \int_a^b x(t) dt \right|^2 \leq (b-a) \int_a^b |x(t)|^2 dt, \quad (1.32)$$

with equality if and only if  $x(t)$  is constant on  $[a, b]$ .

**$C^q([a, b])$ : Spaces of Continuous Functions with  $q$  Continuous Derivatives** For any finite  $a$  and  $b$ , the space  $C([a, b])$  is defined as the space of complex-valued continuous functions over  $[a, b]$  with inner product and norm given in (1.30). The space of complex-valued continuous functions over  $[a, b]$  that are further restricted to have  $q$  continuous derivatives is denoted  $C^q([a, b])$ , so  $C([a, b]) = C^0([a, b])$ . Each  $C^q([a, b])$  space is a subspace of  $\mathbb{C}^{[a, b]}$ . As an example, the set of polynomial functions forms a subspace of  $C^q([a, b])$  for any  $a, b \in \mathbb{R}$  and  $q \in \mathbb{N}$ . This is because the set is closed under the vector space operations and polynomials are infinitely differentiable.

A  $C^q([a, b])$  space is very similar to  $\mathcal{L}^2([a, b])$ . The distinction is completeness (which  $C^q([a, b])$  can lack), which is a defining characteristic of Hilbert spaces that we introduce in the next section.

**Spaces of Random Variables** The set of complex random variables defined in some probabilistic model form a vector space over the complex numbers; all the properties required in Definition 1.1 are inherited from the complex numbers, with the constant 0 as the additive identity.

A useful inner product to define on this vector space is

$$\langle x, y \rangle = E[xy^*]. \quad (1.33)$$

This clearly satisfies properties Definition 1.7(i)–(iii), and also

$$\langle x, x \rangle = E[xx^*] = E[|x|^2] \geq 0.$$

Only the second part of Definition 1.7(iv) is subtle. It is indeed the case that  $E[xx^*] = 0$  implies  $x = 0$ . This is because of the sense of equality for random variables reviewed in Appendix 1.C. The norm induced by this inner product is

$$\|x\| = \sqrt{\langle x, x \rangle} = \sqrt{E[|x|^2]}. \quad (1.34)$$

This shows that if we restrict to random variables with finite second moment,  $E[|x|^2] < \infty$ , we have a normed vector space.

When  $E[x] = 0$ , the norm  $\|x\|$  is the standard deviation of  $x$ . When  $E[y] = 0$  also,  $\langle x, y \rangle$  is the covariance of  $x$  and  $y$ ; the normalized inner product  $\langle x, y \rangle / (\|x\| \|y\|)$  equals the correlation coefficient.

### Standard Normed Vector Spaces

**$\mathbb{C}^N$  Spaces** As we said earlier, we can define other norms on  $\mathbb{C}^N$ . For example, the  $p$  norm is defined as

$$\|x\|_p = \left( \sum_{n=0}^{N-1} |x_n|^p \right)^{1/p}, \quad (1.35a)$$

for  $p \in [1, \infty)$ . Since the sum above has a finite number of terms, there is no doubt that the sums converge. Thus, we take as a vector space of interest the entire  $\mathbb{C}^N$ ; note how this contrasts with some of the examples we see shortly ( $\ell^p(\mathbb{Z})$  spaces).

For  $p = 1$ , this norm is called the *taxicab norm* or *Manhattan norm* because  $\|x\|_1$  represents the driving distance from the origin to  $x$  following a rectilinear street grid. For  $p = 2$ , we get our usual Euclidean square norm from (1.35a), and only in that case is a  $p$  norm induced by an inner product. The natural extension of (1.35a) to  $p = \infty$  (see Exercise 1.14) defines the  $\infty$  norm as:

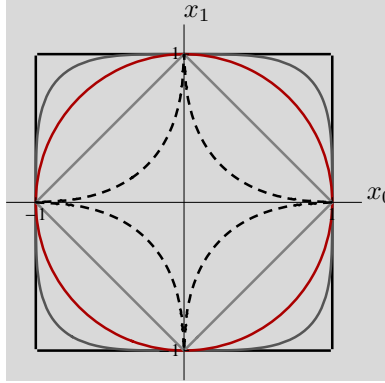
$$\|x\|_\infty = \max(|x_0|, |x_1|, \dots, |x_{N-1}|). \quad (1.35b)$$

Using (1.35a) for  $p \in (0, 1)$  does not give a norm but can still be a useful quantity. The failure to satisfy the requirements of a norm and an interpretation of (1.35a) with  $p \rightarrow 0$  are explored in Exercise 1.15.

All norms on finite-dimensional spaces are equivalent in the sense that any two norms bound each other within constant factors (see Exercise 1.16). This is a crude equivalence that leaves significant differences in which vectors are considered larger than others, and it does not extend to infinite-dimensional spaces. Figure 1.7 shows this pictorially by showing the sets of unit-norm vectors for different  $p$  norms. All vectors ending on the lines have a unit norm in the corresponding  $p$  norm. For example, with the usual Euclidean norm, unit-norm vectors fall on a circle; on the other hand, in 1 norm they fall on the diamond-shaped polygon. Note that only for  $p = 2$  is the set of unit-norm vectors invariant to rotation of the coordinate system.

**$\ell^p(\mathbb{Z})$  Spaces** We can define other norms on  $\mathbb{C}^{\mathbb{Z}}$  as well (like we did for  $\mathbb{C}^N$ ). However, because the space is infinite, the choice of the norm and the requirement that it be finite restricts  $\mathbb{C}^{\mathbb{Z}}$  to a smaller set. For example, for  $p \in [1, \infty)$ , the  $\ell^p$  norm is

$$\|x\|_p = \left( \sum_{n \in \mathbb{Z}} |x_n|^p \right)^{1/p}. \quad (1.36a)$$



**Figure 1.7:** Sets of unit-norm vectors for different  $p$  norms. From outside in:  $p = \infty$  (black),  $p = 4$  (dark gray),  $p = 2$  (red),  $p = 1$  (gray), and  $p = 1/2$  (dashed). (The  $p = \frac{1}{2}$  case is not a norm.) Vectors ending on the lines are of unit norm in the corresponding  $p$  norm.

Analogously to (1.35b), we extend this to the  $\ell^\infty$  norm as

$$\|x\|_\infty = \sup_{n \in \mathbb{Z}} |x_n|. \quad (1.36b)$$

We have already introduced the  $\ell^p$  norm for  $p = 2$  in (1.28); only in that case is an  $\ell^p$  norm induced by an inner product. We can now define the spaces associated with the  $\ell^p$  norms:

**DEFINITION 1.11** ( $\ell^p(\mathbb{Z})$ ) For any  $p \in [1, \infty]$ , the normed vector space  $\ell^p(\mathbb{Z})$  is the subspace of  $\mathbb{C}^{\mathbb{Z}}$  consisting of vectors with finite  $\ell^p$  norm.

Solved Exercise 1.2 shows that the subset of  $\mathbb{C}^{\mathbb{Z}}$  with vectors of finite  $\ell^p(\mathbb{Z})$  norm form a subspace. Since  $\ell^p(\mathbb{Z})$  is defined as a subspace of  $\mathbb{C}^{\mathbb{Z}}$ , it inherits the operations of vector addition and scalar multiplication from  $\mathbb{C}^{\mathbb{Z}}$ . The norm is the  $\ell^p$  norm (1.36).

**EXAMPLE 1.8** Consider the sequence  $x$  given by

$$x_n = \begin{cases} 0, & n \leq 0; \\ 1/n^a, & n > 0, \end{cases}$$

for some real number  $a \geq 0$ . Let us determine which of the spaces  $\ell^p(\mathbb{Z})$  contain  $x$ . To check whether  $x$  is in  $\ell^p(\mathbb{Z})$  for  $p \in [1, \infty)$ , we need to determine whether

$$\|x\|_p^p = \sum_{n=1}^{\infty} \left| \frac{1}{n^a} \right|^p = \sum_{n=1}^{\infty} \frac{1}{n^{pa}}$$



## 1.2. Vector Spaces

33

converges. The necessary and sufficient condition for convergence is  $pa > 1$ , so we conclude that  $x \in \ell^p(\mathbb{Z})$  for  $p > 1/a$  and  $a > 0$ . For  $a = 0$ , the above does not converge. For  $x \in \ell^\infty(\mathbb{Z})$ ,  $x$  must be bounded, which occurs for all  $a \geq 0$ .

This example illustrates a simple inclusion property proven as Exercise 1.17:

$$p < q \quad \text{implies} \quad \ell^p(\mathbb{Z}) \subset \ell^q(\mathbb{Z}). \quad (1.37)$$

This can loosely be visualized with Figure 1.7: the larger the value of  $p$ , the larger the set of vectors with a given norm. In particular,  $\ell^1(\mathbb{Z}) \subset \ell^2(\mathbb{Z})$ . In other words, if a sequence has a finite  $\ell^1$  norm, then it has a finite  $\ell^2$  norm. Beware that the opposite is not true; if a sequence has a finite  $\ell^2$  norm, it does not follow that it has a finite  $\ell^1$  norm.

EXAMPLE 1.9 Consider the sequence  $x_n = 1/n$ , for  $n \in \mathbb{Z}^+$ :

$$\|x\|_2^2 = \sum_{n=1}^{\infty} \left| \frac{1}{n} \right|^2 = \frac{1}{6}\pi^2 \text{ converges, while } \|x\|_1 = \sum_{n=1}^{\infty} \left| \frac{1}{n} \right| \text{ diverges.}$$

Thus,  $x \in \ell^2(\mathbb{Z})$  and  $x \notin \ell^1(\mathbb{Z})$ .

**$\mathcal{L}^p(\mathbb{R})$  Spaces** Like for sequences, we can define other norms on  $\mathbb{C}^{\mathbb{R}}$  as well. Again, because the space is infinite, the choice of the norm and the requirement that it be finite restricts  $\mathbb{C}^{\mathbb{R}}$  to a smaller set. For example, for  $p \in [1, \infty)$ , the  $\mathcal{L}^p$  norm is

$$\|x\|_p = \left( \int_{-\infty}^{\infty} |x(t)|^p dt \right)^{1/p}. \quad (1.38a)$$

The extension to  $p = \infty$  leads to the  $\mathcal{L}^\infty$  norm as

$$\|x\|_\infty = \operatorname{ess\,sup}_{t \in \mathbb{R}} |x(t)|. \quad (1.38b)$$

We have already introduced the  $\mathcal{L}^p$  norm for  $p = 2$  in (1.29); only in that case is an  $\mathcal{L}^p$  norm induced by an inner product. We can now define the spaces associated with the  $\mathcal{L}^p$  norms:

DEFINITION 1.12 ( $\mathcal{L}^p(\mathbb{R})$ ) For any  $p \in [1, \infty]$ , the normed vector space  $\mathcal{L}^p(\mathbb{R})$  is the subspace of  $\mathbb{C}^{\mathbb{R}}$  consisting of vectors with finite  $\mathcal{L}^p$  norm.

Since  $\mathcal{L}^p(\mathbb{R})$  is defined as a subspace of  $\mathbb{C}^{\mathbb{R}}$ , it inherits the operations of vector addition and scalar multiplication from  $\mathbb{C}^{\mathbb{R}}$ . The norm is the  $\mathcal{L}^p$  norm (1.38).

One can also use the same norms on different domains; for example, we can define the domain to be  $[a, b]$  and use a finite  $\mathcal{L}^p$  norm on it to yield the space  $\mathcal{L}^p([a, b])$  (like we did for  $\mathcal{L}^2(\mathbb{R})$  and  $\mathcal{L}^2([a, b])$ ). We will use this in Chapters 2 and 3, where we will often be operating on  $\mathcal{L}^p([-\pi, \pi])$ .

## 1.3 Hilbert Spaces

We are going to do most of our work in Hilbert spaces. These are inner product spaces seen in the previous section, with the additional requirement of *completeness*. Completeness is somewhat technical, and for a basic understanding it will suffice to have faith that we work in vector spaces of sequences and functions in which convergence makes sense. We will furthermore be mostly concerned with *separable* Hilbert spaces because these spaces have countable bases.

### 1.3.1 Convergence

Convergence of sequences of numbers should be a familiar concept; it is reviewed in Appendix 1.A.2. Convergence of a sequence of vectors requires a metric, and we limit our attention to metrics induced by norms.

**DEFINITION 1.13 (CONVERGENT SEQUENCE OF VECTORS)** A sequence of vectors  $x_0, x_1, \dots$  in a normed vector space  $V$  is said to converge to  $v \in V$  when  $\lim_{k \rightarrow \infty} \|v - x_k\| = 0$ . In other words: Given any  $\varepsilon > 0$ , there exists a  $K_\varepsilon$  such that

$$\|v - x_k\| < \varepsilon \quad \text{for all } k > K_\varepsilon.$$

The elements of a convergent sequence eventually stay arbitrarily close to  $v$ . Not only does the definition of convergence of sequences of vectors use a norm, but whether a sequence converges can depend on the choice of norm. This is illustrated in the following example:

**EXAMPLE 1.10 (CONVERGENCE IN DIFFERENT NORMS)**

(i) For each  $k \in \mathbb{Z}^+$ , let

$$x_k(t) = \begin{cases} 1, & \text{for } t \in [0, 1/k]; \\ 0, & \text{otherwise.} \end{cases}$$

Also, let  $v(t) = 0$  for all  $t$ . For any  $p \in [1, \infty)$ , using the expression for the  $\mathcal{L}^p$  norm, (1.38a),

$$\|v - x_k\|_p = \left( \int_{-\infty}^{\infty} |v(t) - x_k(t)|^p dt \right)^{1/p} = \left( \frac{1}{k} \right)^{1/p} \xrightarrow{k \rightarrow \infty} 0,$$

so  $x_1, x_2, \dots$  converges to  $v$ . For  $p = \infty$ , using the expression for the  $\mathcal{L}^\infty$  norm, (1.38b),  $\|v - x_k\|_\infty = 1$  for all  $k$ , so the sequence does not converge to  $v$  under the  $\mathcal{L}^\infty$  norm.

(ii) Let  $\alpha \in (0, 1)$ , and for each  $k \in \mathbb{Z}^+$ , let

$$x_{k,n} = \begin{cases} 1/k^\alpha, & \text{for } n \in \{1, 2, \dots, k\}; \\ 0, & \text{otherwise.} \end{cases}$$

## 1.3. Hilbert Spaces

35

Also, let  $v_n = 0$  for all  $n$ . Using the expression for the  $\ell^p$  norm, (1.36a),

$$\|v - x_k\|_p = \left( \sum_{n=1}^k (1/k^\alpha)^p \right)^{1/p} = \left( \frac{1}{k} \right)^{(\alpha p - 1)/p},$$

so  $x_1, x_2, \dots$  converges to  $v$  when  $p \in (1/\alpha, \infty)$ . For  $p = \infty$ , using the expression for the  $\ell^\infty$  norm, (1.36b),  $\|v - x_k\|_\infty = 1/k$  for all  $k$ , so the sequence converges to  $x$  under the  $\ell^\infty$  norm as well.

As reviewed in Appendix 1.A.1, a set of real numbers is closed if and only if it contains the limits of all its convergent sequences; for example,  $(0, 1]$  is not closed in  $\mathbb{R}$  since the sequence of numbers  $x_k = 1/k$ ,  $k \in \mathbb{Z}^+$ , lies in  $(0, 1]$  and converges, but its limit point 0 is not in  $(0, 1]$ . Carrying this over to Hilbert space setting yields the following:

**DEFINITION 1.14 (CLOSED SUBSPACE)** A subspace  $S$  of a normed vector space  $V$  is called closed when it contains all limits of sequences of vectors in  $S$ .

Subspaces of finite-dimensional normed vector spaces are always closed. Exercise 1.19 gives an example of a subspace of an infinite-dimensional normed vector space that is not closed.

Subspaces often arise as the span of set of vectors. As the following example shows, the span of an infinite set of vectors is not necessarily closed. For this reason, we frequently work with the closure of a span.

**EXAMPLE 1.11 (SPAN MAY NOT BE CLOSED)** Consider the following infinite set of vectors from  $\ell^2(\mathbb{N})$ : for each  $k \in \mathbb{N}$ , let the sequence  $s_k$  be 0 except for a 1 in the  $k$ th position. Recall from Definition 1.4 that the span is all finite linear combinations, even when the set of vectors is infinite. Thus,  $\text{span}(\{s_0, s_1, \dots\})$  is the subspace of all vectors in  $\ell^2(\mathbb{Z}^+)$  that have a finite number of nonzero entries. To prove that the span is not closed, we must find a sequence of vectors in the span (each having finitely-many nonzero entries) that converges to a vector not in the span (having infinitely-many nonzero entries). For example, let  $v$  be any sequence in  $\ell^2(\mathbb{N})$  with infinite support, for example  $v_n = 1/(n+1)$  for  $n \in \mathbb{N}$ . Then for each  $k \in \mathbb{N}$ , define vector  $x_k \in \ell^2(\mathbb{N})$  by

$$x_{k,n} = \begin{cases} v_n, & \text{for } n = 0, 1, \dots, k; \\ 0, & \text{otherwise.} \end{cases}$$

For each  $k \in \mathbb{N}$ , the vector  $x_k$  is a linear combination of  $\{s_0, s_1, \dots, s_k\}$ . While the sequence  $x_0, x_1, \dots$  converges to  $v$  (under the  $\ell^2(\mathbb{N})$  norm), its limit  $v$  is not in the span.

Since the closure of a set is the set of all limit points of convergent sequences in the set, the closure of the span of an infinite set of vectors is the set of all convergent



**Figure 1.8:** The first few partial sums of  $\sum_{n=0}^{\infty} 1/n!$ , each rational, converging to the irrational number  $e$ .

infinite linear combinations:

$$\overline{\text{span}}(\{\varphi_k\}_{k \in \mathbb{Z}}) = \left\{ \sum_{k \in \mathbb{Z}} \alpha_k \varphi_k \mid \alpha_k \in \mathbb{C} \text{ and the sum converges} \right\}.$$

The closure of the span of a set of vectors is always a closed subspace.

### 1.3.2 Completeness

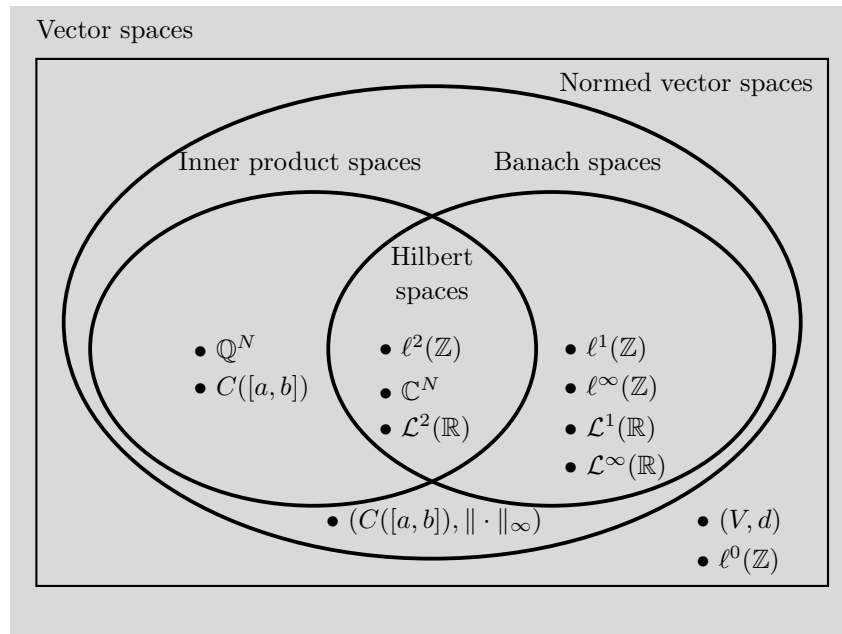
It is awkward to do analysis in the set of rational numbers  $\mathbb{Q}$  instead of in  $\mathbb{R}$  because  $\mathbb{Q}$  has infinitesimal gaps that can be limit points of sequences in  $\mathbb{Q}$ . A familiar example is that  $x_k = \sum_{n=0}^k 1/n!$  is rational for every nonnegative integer  $k$ , but the limit of the sequence is the irrational number  $e$  (see Figure 1.8). If we want the limit of any sequence to make sense, we need to work in  $\mathbb{R}$ , which is the *completion* of  $\mathbb{Q}$ . Working only in  $\mathbb{Q}$ , it would be hard to distinguish between sequences that converge to an irrational number and ones that do not converge at all—neither would have a limit point in the space.

Complete vector spaces are those in which sequences that intuitively ought to converge (the Cauchy sequences) have a limit in the same space.

**DEFINITION 1.15 (CAUCHY SEQUENCE OF VECTORS)** A sequence of vectors  $x_0, x_1, \dots$  in a normed vector space is called a Cauchy sequence when: Given any  $\varepsilon > 0$ , there exists a  $K_\varepsilon$  such that

$$\|x_k - x_m\| < \varepsilon \quad \text{for all } k, m > K_\varepsilon.$$

The elements of a Cauchy sequence eventually stay arbitrarily close to each other. Thus it may be intuitive that a Cauchy sequence must converge; this is in fact true for real-valued sequences. However this is not true in all vector spaces, and it gives us important terminology:



**Figure 1.9:** Relationships between types of vector spaces. Several examples of vector spaces are marked. For  $\mathbb{Q}^N$  and  $\mathbb{C}^N$  we assume the standard inner product.  $(V, d)$  represents any vector space with the discrete metric as described in Exercise 1.13.  $(C([a, b]), \|\cdot\|_\infty)$  represents  $C([a, b])$  with the  $\mathcal{L}^2$  norm replaced by the  $\mathcal{L}^\infty$  norm.  $\ell^0(\mathbb{Z})$  is described in Exercise 1.19.

**DEFINITION 1.16 (COMPLETENESS AND HILBERT SPACE)** A normed vector space  $V$  is said to be complete when every Cauchy sequence in  $V$  converges to a vector in  $V$ . A complete inner product space is called a Hilbert space.

A complete normed vector space is called a *Banach space*.

**EXAMPLE 1.12** Ignoring for the moment that Definition 1.1 restricts the set of scalars to  $\mathbb{R}$  or  $\mathbb{C}$ , consider  $\mathbb{Q}$  as a normed vector space over the scalars  $\mathbb{Q}$ , with ordinary addition and multiplication and norm  $\|x\| = |x|$ . This vector space is not complete because there exist rational sequences with irrational limits, such as the example of the number  $e$  we have just seen (see Figure 1.8).

### Standard Spaces

From Definition 1.16, completeness makes sense only in a normed vector space. We now comment on the completeness of the standard spaces we discussed in Sec-

tion 1.2.4 (see Figure 1.9).

- (i) All finite-dimensional spaces are complete.<sup>10</sup> For example,  $\mathbb{C}$  as a normed vector space over  $\mathbb{C}$  with ordinary addition and multiplication and with norm  $\|x\| = |x|$  is complete. This can be used to show that  $\mathbb{C}^N$  is complete under ordinary addition and multiplication and with any  $p$  norm; see Exercise 1.22.  $\mathbb{C}^N$  under the 2 norm is a Hilbert space.
- (ii) All  $\ell^p(\mathbb{Z})$  spaces are complete; in particular,  $\ell^2(\mathbb{Z})$  is a Hilbert space.
- (iii) All  $\mathcal{L}^p(\mathbb{R})$  spaces are complete; in particular,  $\mathcal{L}^2(\mathbb{R})$  is a Hilbert space. An  $\mathcal{L}^p$  space can either be understood to be complete because of Lebesgue measurability and the use of Lebesgue integration, or it can be taken as the completion of the space of continuous functions with finite  $\mathcal{L}^p$  norm.
- (iv)  $C^q([a, b])$  spaces are not complete, except under the  $\mathcal{L}^\infty$  norm. For example,  $C([0, 1])$  is not complete as there exist Cauchy sequences of continuous functions whose limits are discontinuous (and hence not in  $C([0, 1])$ ); see Solved Exercise 1.3.
- (v) We consider spaces of random variables only under the inner product (1.33) and norm (1.34). These inner product spaces are complete and hence Hilbert spaces.

**Separability** Separability is more technical than completeness. A space is called *separable* when it contains a countable dense subset. For example,  $\mathbb{R}$  is separable since  $\mathbb{Q}$  is dense in  $\mathbb{R}$  and is countable. However, these topological properties are not of much interest here.

We are interested in separable Hilbert spaces because a Hilbert space contains a countable basis if and only if it is separable (we formally define a basis in Section 1.5). The Hilbert spaces that we will use frequently (as marked in Figure 1.9) are all separable. Also, a closed subspace of a separable Hilbert space is separable, so it contains a countable basis as well.

### 1.3.3 Linear Operators

Having dispensed with technicalities, we are now ready to develop operational Hilbert space machinery. We start with linear operators, which generalize finite-dimensional matrix multiplication (see Appendix 1.B for more details).

**DEFINITION 1.17 (LINEAR OPERATOR)** A function  $A : H_0 \rightarrow H_1$  is called a linear operator from  $H_0$  to  $H_1$  when, for all  $x, y$  in  $H_0$  and  $\alpha$  in  $\mathbb{C}$  (or  $\mathbb{R}$ ):

- (i) *Additivity*:  $A(x + y) = Ax + Ay$ .
- (ii) *Scalability*:  $A(\alpha x) = \alpha(Ax)$ .

<sup>10</sup>Recall the restriction of the set of scalars to  $\mathbb{R}$  or  $\mathbb{C}$ . Without this restriction, there are finite-dimensional vector spaces that are not complete, as in Example 1.12.

## 1.3. Hilbert Spaces

39

When domain  $H_0$  and codomain  $H_1$  are the same,  $A$  is also called a linear operator on  $H_0$ .

Note the convention of writing  $Ax$  instead of  $A(x)$ , just as is done for matrix multiplication. In fact, linear operators from  $\mathbb{C}^N$  to  $\mathbb{C}^M$  and matrices in  $\mathbb{C}^{M \times N}$  are exactly the same thing.

Many concepts from finite-dimensional linear algebra extend to linear operators on Hilbert spaces in rather obvious ways. For example, the *null space* or *kernel* of a linear operator  $A : H_0 \rightarrow H_1$  is the subspace of  $H_0$  that  $A$  maps to  $\mathbf{0}$ :

$$\mathcal{N}(A) = \{x \in H_0 \mid Ax = \mathbf{0}\}. \quad (1.39)$$

The *range* of a linear operator  $A : H_0 \rightarrow H_1$  is a subspace of  $H_1$ :

$$\mathcal{R}(A) = \{Ax \in H_1 \mid x \in H_0\}. \quad (1.40)$$

Some generalizations to Hilbert spaces are simplified by limiting attention to bounded linear operators.

**DEFINITION 1.18 (OPERATOR NORM AND BOUNDED LINEAR OPERATOR)** The operator norm of  $A$ , denoted by  $\|A\|$ , is defined as

$$\|A\| = \sup_{\|x\|=1} \|Ax\|. \quad (1.41)$$

A linear operator is called bounded when its operator norm is finite.

It is implicit in the definition that  $\|x\|$  uses the norm of  $H_0$  and  $\|Ax\|$  uses the norm of  $H_1$ . This concept applies equally well with  $H_0$  and  $H_1$  replaced by any normed vector spaces  $V_0$  and  $V_1$ .

**EXAMPLE 1.13 (UNBOUNDED OPERATOR)** Consider  $A : \ell^2(\mathbb{Z}) \rightarrow \ell^2(\mathbb{Z})$  defined by

$$(Ax)_n = |n| x_n \quad \text{for all } n \in \mathbb{Z}.$$

While this is a linear operator, it is not bounded. To see this by contradiction, suppose the operator norm  $\|A\|$  is finite. Then there is an integer  $M$  larger than  $\|A\|$ , and by considering as input the sequence that is 0 except for a 1 in the  $M$ th position, we obtain a contradiction.

Linear operators with finite-dimensional codomains are always bounded. Conversely, by limiting attention to bounded linear operators we are able to extend concepts to Hilbert spaces while maintaining most intuitions from finite-dimensional linear algebra. For example, bounded linear operators are continuous:

$$\text{if } x_k \xrightarrow{k \rightarrow \infty} v \text{ then } Ax_k \xrightarrow{k \rightarrow \infty} Av.$$

DEFINITION 1.19 (INVERSE) A bounded linear operator  $A : H_0 \rightarrow H_1$  is called invertible if there exists a bounded linear operator  $B : H_1 \rightarrow H_0$  such that

$$BAx = x, \quad \text{for every } x \text{ in } H_0, \quad \text{and} \quad (1.42a)$$

$$AB y = y, \quad \text{for every } y \text{ in } H_1. \quad (1.42b)$$

When such a  $B$  exists, it is unique, is denoted by  $A^{-1}$ , and is called the *inverse* of  $A$ ;  $B$  is called a *left inverse* of  $A$  when (1.42a) holds and  $B$  is called a *right inverse* of  $A$  when (1.42b) holds.

EXAMPLE 1.14 (LINEAR OPERATORS)

(i) Ordinary matrix multiplication by the matrix

$$A = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix}$$

defines a linear operator on  $\mathbb{R}^2$ ,  $A : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ . It is bounded, and its operator norm (assuming the 2 norm for both the domain and the codomain) is 4. We show here how to obtain the norm of  $A$  by direct computation (we could also use the relationship between eigenvalues, singular values, and the operator norm, explored in Exercise 1.67):

$$\begin{aligned} \|A\| &= \sup_{\|x\|=1} \|Ax\| = \sup_{\theta} \left\| \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} \right\| = \sup_{\theta} \left\| \begin{bmatrix} 3 \cos \theta + \sin \theta \\ \cos \theta + 3 \sin \theta \end{bmatrix} \right\| \\ &= \sup_{\theta} \sqrt{(3 \cos \theta + \sin \theta)^2 + (\cos \theta + 3 \sin \theta)^2} \\ &= \sup_{\theta} \sqrt{10 \cos^2 \theta + 10 \sin^2 \theta + 12 \sin \theta \cos \theta} \\ &= \sup_{\theta} \sqrt{10 + 6 \sin 2\theta} = 4. \end{aligned}$$

The null space of  $A$  is only the vector  $\mathbf{0}$ , the range of  $A$  is all of  $\mathbb{R}^2$ , and

$$A^{-1} = \begin{bmatrix} 3/8 & -1/8 \\ -1/8 & 3/8 \end{bmatrix}.$$

(ii) Ordinary matrix multiplication by the matrix

$$A = \begin{bmatrix} 1 & j & 0 \\ 1 & 0 & j \end{bmatrix}$$

defines a linear operator  $A : \mathbb{C}^3 \rightarrow \mathbb{C}^2$ . It is bounded, and its operator norm (assuming the 2 norm for both the domain and the codomain) is  $\sqrt{3}$ . Its null space is  $\{[x_0 \ jx_0 \ jx_0]^T\}$ , its range is all of  $\mathbb{C}^2$ , and it is not invertible. (There exists  $B$  satisfying (1.42b), but no  $B$  can satisfy (1.42a).)



## 1.3. Hilbert Spaces

41

- (iii) For some fixed complex-valued sequence  $(\alpha_k)_{k \in \mathbb{Z}}$ , consider the component-wise multiplication

$$(Ax)_k = \alpha_k x_k \quad (1.43a)$$

as a linear operator on  $\ell^2(\mathbb{Z})$ . We can write this with infinite vectors and matrices as

$$Ax = \begin{bmatrix} \ddots & 0 & & & \\ 0 & \alpha_{-1} & 0 & & \\ & 0 & \boxed{\alpha_0} & 0 & \\ & & 0 & \alpha_1 & 0 \\ & & & 0 & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ \vdots \end{bmatrix}. \quad (1.43b)$$

It is easy to check that Definition 1.17(i) and (ii) are satisfied, but we must constrain  $\alpha$  to ensure that the result is in  $\ell^2(\mathbb{Z})$ . For example,  $\|\alpha\|_\infty = M < \infty$  ensures that  $Ax$  is in  $\ell^2(\mathbb{Z})$  for any  $x$  in  $\ell^2(\mathbb{Z})$ . Furthermore, the operator is bounded and  $\|A\| = M$ . The operator is invertible when  $\inf_k |\alpha_k| > 0$ . In this case, the inverse is given by

$$A^{-1}y = \begin{bmatrix} \ddots & 0 & & & \\ 0 & 1/\alpha_{-1} & 0 & & \\ & 0 & \boxed{1/\alpha_0} & 0 & \\ & & 0 & 1/\alpha_1 & 0 \\ & & & 0 & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ \vdots \end{bmatrix}.$$

**Adjoint Operator** Finite-dimensional linear algebra has many uses for transposes and conjugate transposes. The conjugate transpose (or Hermitian transpose) is generalized by the adjoint of an operator.

**DEFINITION 1.20 (ADJOINT AND SELF-ADJOINT OPERATORS)** The linear operator  $A^* : H_1 \rightarrow H_0$  is called the adjoint of the linear operator  $A : H_0 \rightarrow H_1$  when

$$\langle Ax, y \rangle_{H_1} = \langle x, A^*y \rangle_{H_0}, \quad \text{for every } x \text{ in } H_0 \text{ and } y \text{ in } H_1. \quad (1.44)$$

When  $A = A^*$ , the operator  $A$  is called self-adjoint or Hermitian.

Note that the adjoint gives a third meaning to  $*$ , the first two being the complex conjugate of a scalar and the Hermitian transpose of a matrix. These meanings are consistent, as we verify in the first two parts of the following example.

**EXAMPLE 1.15 (ADJOINT OPERATORS)**

- (i) For any Hilbert space  $H$ , consider  $A : H \rightarrow H$  given by  $Ax = \alpha x$  for some scalar  $\alpha$ . For any  $x$  and  $y$  in  $H$ ,

$$\langle Ax, y \rangle = \langle \alpha x, y \rangle \stackrel{(a)}{=} \alpha \langle x, y \rangle \stackrel{(b)}{=} \langle x, \alpha^* y \rangle,$$

where (a) follows from linearity in the first argument of the inner product; and (b) from conjugate linearity in the second argument of the inner product. Comparing to (1.44), the adjoint of  $A$  is  $A^* y = \alpha^* y$ . Put simply: the adjoint of multiplication by a scalar is multiplication by the conjugate of the scalar, consistent with using  $*$  for conjugation of a scalar.

- (ii) Consider a linear operator  $A : \mathbb{C}^N \rightarrow \mathbb{C}^M$ . The  $\mathbb{C}^N$  and  $\mathbb{C}^M$  inner products can both be written as  $\langle x, y \rangle = y^* x$ , where  $*$  represents the Hermitian transpose. Thus for any  $x \in \mathbb{C}^N$  and  $y \in \mathbb{C}^M$ ,

$$\langle Ax, y \rangle_{\mathbb{C}^M} = y^*(Ax) = (y^*A)x \stackrel{(a)}{=} (A^*y)^*x = \langle x, A^*y \rangle_{\mathbb{C}^N},$$

where in (a) we use  $A^*$  to represent the Hermitian transpose of the matrix  $A$ . Comparing to (1.44), it seems we have reached a tautology, but this is because the use of  $A^*$  as the adjoint of linear operator  $A$  and the Hermitian transpose of matrix  $A$  are consistent. Put simply: the adjoint of multiplication by a matrix is multiplication by the Hermitian transpose of the matrix, consistent with using  $*$  for Hermitian transpose of a matrix.

- (iii) Consider the linear operator defined in (1.43). For any  $x$  and  $y$  in  $\ell^2(\mathbb{Z})$ ,

$$\langle Ax, y \rangle_{\ell^2} \stackrel{(a)}{=} \sum_{n \in \mathbb{Z}} (\alpha_n x_n) y_n^* \stackrel{(b)}{=} \sum_{n \in \mathbb{Z}} x_n (\alpha_n^* y_n)^*$$

where (a) follows from (1.43) and the definition of the  $\ell^2(\mathbb{Z})$  inner product, (1.20b); and (b) from commutativity and associativity of scalar multiplication along with  $(\alpha_n^*)^* = \alpha_n$ . Our goal in expanding  $\langle Ax, y \rangle$  above is to see the result as the inner product between  $x$  and some linear operator applied to  $y$ . Comparing the final expression to the definition of the  $\ell^2(\mathbb{Z})$  inner product, the componentwise multiplication  $(A^*y)_n = \alpha_n^* y_n$  defines the adjoint.

The above examples are amongst the simplest, and they do not necessarily clearly reveal the role of an adjoint. In Hilbert spaces, the relationships between vectors are measured by inner products. The defining relation of the adjoint (1.44) shows that the action of  $A$  on  $H_0$  is mimicked by the action of the adjoint  $A^*$  on  $H_1$ ; this mimicry is only visible through the applicable inner products, so the adjoint itself depends on these inner products. Loosely, if  $A$  has some effect,  $A^*$  preserves the geometry of that effect while acting with reversed domain and codomain.

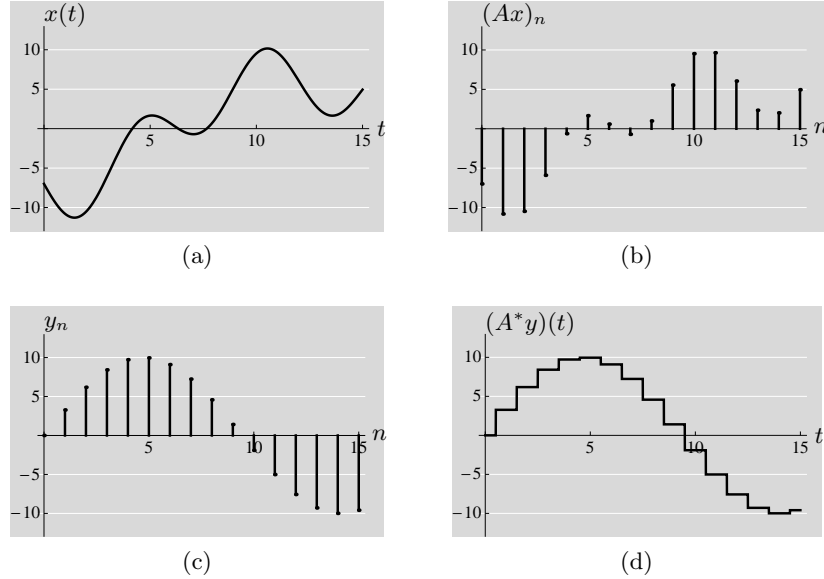
In general, finding an adjoint requires some ingenuity, as we now show:

EXAMPLE 1.16 (LOCAL AVERAGING AND ITS ADJOINT) The operator

$$(Ax)_n = \int_{n-1/2}^{n+1/2} x(t) dt \tag{1.45}$$

## 1.3. Hilbert Spaces

43



**Figure 1.10:** Illustration of the adjoint of an operator. (a) We start with a function  $x$  in  $\mathcal{L}^2(\mathbb{R})$ . (b) The local averaging operator  $A$  in (1.45) gives a sequence in  $\ell^2(\mathbb{Z})$ . (c)  $y$  is an arbitrary sequence in  $\ell^2(\mathbb{Z})$ . (d) The adjoint  $A^*$  is a linear operator from  $\ell^2(\mathbb{Z})$  to  $\mathcal{L}^2(\mathbb{R})$  that uniquely preserves geometry in that  $\langle Ax, y \rangle_{\ell^2} = \langle x, A^*y \rangle_{\mathcal{L}^2}$ . The adjoint of local averaging is to form a piecewise-constant function as in (1.48).

takes local averages of the function  $x(t)$  to yield a sequence  $(Ax)_n$ . (This operation is depicted in Figure 1.10 and is a form of sampling, covered in detail in Chapter 4.) We will first verify that  $A$  is a linear operator from  $\mathcal{L}^2(\mathbb{R})$  to  $\ell^2(\mathbb{Z})$  and then find its adjoint.

The operator  $A$  clearly satisfies Definition 1.17(i) and (ii); we just need to be sure that the result is in  $\ell^2(\mathbb{Z})$ . Given  $x \in \mathcal{L}^2(\mathbb{R})$ , let us compute the  $\ell^2$  norm of  $Ax$ :

$$\begin{aligned} \|Ax\|_{\ell^2}^2 &\stackrel{(a)}{=} \sum_{n \in \mathbb{Z}} |(Ax)_n|^2 \stackrel{(b)}{=} \sum_{n \in \mathbb{Z}} \left| \int_{n-1/2}^{n+1/2} x(t) dt \right|^2 \\ &\stackrel{(c)}{\leq} \sum_{n \in \mathbb{Z}} \int_{n-1/2}^{n+1/2} |x(t)|^2 dt = \int_{-\infty}^{\infty} |x(t)|^2 dt = \|x\|_{\mathcal{L}^2}^2, \end{aligned}$$

where (a) follows from the definition of the  $\ell^2$  norm (1.28); (b) from (1.45); and (c) from (1.32). Thus,  $Ax$  is indeed in  $\ell^2(\mathbb{Z})$  since its norm is bounded by  $\|x\|_{\mathcal{L}^2}$ , which we know is finite since  $x \in \mathcal{L}^2(\mathbb{R})$ .

We now derive the adjoint of the operator (1.45). To do this, we must find an operator  $A^* : \ell^2(\mathbb{Z}) \rightarrow \mathcal{L}^2(\mathbb{R})$  such that  $\langle Ax, y \rangle_{\ell^2} = \langle x, A^*y \rangle_{\mathcal{L}^2}$  for any  $x \in \mathcal{L}^2(\mathbb{R})$  and  $y \in \ell^2(\mathbb{Z})$ . After expanding both expressions using the definitions

of the two inner products, the unique choice for  $A^*y$  will be clear:

$$\begin{aligned} \langle Ax, y \rangle_{\ell^2} &\stackrel{(a)}{=} \sum_{n \in \mathbb{Z}} (Ax)_n y_n^* \stackrel{(b)}{=} \sum_{n \in \mathbb{Z}} \left( \int_{n-1/2}^{n+1/2} x(t) dt \right) y_n^* \\ &\stackrel{(c)}{=} \sum_{n \in \mathbb{Z}} \int_{n-1/2}^{n+1/2} x(t) y_n^* dt \end{aligned} \quad (1.46)$$

where (a) follows from the definition of an inner product in  $\ell^2(\mathbb{Z})$ , (1.28); (b) from (1.45); and in (c) we pull  $y_n$  into the integral since it does not depend on  $t$ . For this final expression to match

$$\langle x, A^*y \rangle_{\mathcal{L}^2} = \int_{-\infty}^{\infty} x(t) ((A^*y)(t))^* dt \quad (1.47)$$

for arbitrary  $x$  and  $y$ , we must define  $A^*y$  as the piecewise-constant function

$$(A^*y)(t) = y_n \quad \text{for } t \in [n - \frac{1}{2}, n + \frac{1}{2}). \quad (1.48)$$

Then the integral in (1.47) breaks into the sum of integrals in (1.46).

The following theorem summarizes several key properties of the adjoint:

**THEOREM 1.21 (ADJOINT PROPERTIES)** Let  $A : H_0 \rightarrow H_1$  be a bounded linear operator.

- (i) The adjoint  $A^*$ , defined through (1.44), exists.
- (ii) The adjoint  $A^*$  is unique.
- (iii) The adjoint of  $A^*$  equals the original operator,  $(A^*)^* = A$ .
- (iv) The operators  $AA^*$  and  $A^*A$  are self-adjoint.
- (v) The operator norms of  $A$  and  $A^*$  are equal,  $\|A^*\| = \|A\|$ .
- (vi) If  $A$  is invertible, the adjoint of the inverse and the inverse of the adjoint are equal,  $(A^{-1})^* = (A^*)^{-1}$ .
- (vii) Let  $B : H_0 \rightarrow H_1$  be a bounded linear operator. Then  $(A + B)^* = A^* + B^*$ .
- (viii) Let  $B : H_1 \rightarrow H_2$  be a bounded linear operator. Then  $(BA)^* = A^*B^*$ .

*Proof.* Parts (i) and (v) are the most technically challenging and go beyond our scope; proofs based on the Riesz representation theorem can be found in texts such as [93]. Parts (ii) and (iii) are proven below, with the remaining parts left for Exercise 1.26.

(ii) Suppose  $B$  and  $C$  are adjoints of  $A$ . Then for any  $x$  in  $H_0$  and  $y$  in  $H_1$ ,

$$0 \stackrel{(a)}{=} \langle x, By \rangle - \langle x, Cy \rangle \stackrel{(b)}{=} \langle x, By - Cy \rangle \stackrel{(c)}{=} \langle x, (B - C)y \rangle,$$

where (a) follows from (1.44); (b) from distributivity of the inner product; and (c) from additivity of the operators. Since this holds for every  $x$  in  $H_0$ , it in particular holds for  $x = (B - C)y$ . By the positive definiteness of the inner product, we must have  $(B - C)y = 0$  for every  $y$  in  $H_1$ . This implies  $By = Cy$  for every  $y$  in  $H_1$ , so the adjoint is unique.

## 1.3. Hilbert Spaces

45

(iii) For any  $x$  in  $H_1$  and  $y$  in  $H_0$ ,

$$\langle A^*x, y \rangle \stackrel{(a)}{=} \langle y, A^*x \rangle^* \stackrel{(b)}{=} \langle Ay, x \rangle^* \stackrel{(c)}{=} \langle x, Ay \rangle,$$

where (a) follows from Hermitian symmetry of the inner product; (b) from (1.44); and (c) from Hermitian symmetry of the inner product.

The adjoint of a bounded linear operator provides key relationships between subspaces (Figure 1.36 in Appendix 1.B illustrates the case when the operator is a finite-dimensional matrix):

$$\mathcal{R}(A)^\perp = \mathcal{N}(A^*), \quad (1.49a)$$

$$\overline{\mathcal{R}(A)} = \mathcal{N}(A^*)^\perp. \quad (1.49b)$$

To see that  $\mathcal{N}(A^*) \subseteq \mathcal{R}(A)^\perp$ , first let  $y \in \mathcal{N}(A^*)$  and  $y' \in \mathcal{R}(A)$ . Then since  $y' = Ax$  for some  $x$ , we can compute

$$\langle y', y \rangle = \langle Ax, y \rangle = \langle x, A^*y \rangle = \langle x, 0 \rangle = 0,$$

which shows  $y \perp \mathcal{R}(A)$ . Conversely, to see that  $\mathcal{R}(A)^\perp \subseteq \mathcal{N}(A^*)$ , let  $y \in \mathcal{R}(A)^\perp$  and let  $x$  be any vector in the domain of  $A$ . Then since

$$0 = \langle Ax, y \rangle = \langle x, A^*y \rangle$$

and choosing  $x = A^*y$  implies  $A^*y = 0$  by positive definiteness of the inner product, we must have  $y \in \mathcal{N}(A^*)$ . These arguments prove (1.49a). The subtleties of infinite dimensions, for example that the range of a bounded linear operator may not be closed, make proving (1.49b) a bit more difficult. Linear operators that arise in later chapters have closed ranges.

**Unitary Operators** Unitary operators are important because they preserve geometry (lengths and angles) when mapping one Hilbert space to another.

**DEFINITION 1.22 (UNITARY OPERATORS)** A bounded linear operator  $A : H_0 \rightarrow H_1$  is called unitary when

- (i) it is *invertible*; and
- (ii) it *preserves inner products*,

$$\langle Ax, Ay \rangle_{H_1} = \langle x, y \rangle_{H_0} \quad \text{for every } x, y \text{ in } H_0. \quad (1.50)$$

Preservation of inner products leads to preservation of norms:

$$\|Ax\|^2 = \langle Ax, Ax \rangle = \langle x, x \rangle = \|x\|^2. \quad (1.51)$$

In Fourier theory, this is called *Parseval's equality*; it is used extensively in the book.

The following theorem provides conditions equivalent to the definition of a unitary operator. These conditions are reminiscent of the standard definition of a unitary matrix.

**THEOREM 1.23 (UNITARY LINEAR OPERATOR)** A bounded linear operator  $A : H_0 \rightarrow H_1$  is unitary if and only if  $A^{-1} = A^*$ .

*Proof.* Condition (1.50) is equivalent to  $A^*$  being a left inverse of  $A$ :

$$A^*A = I \quad \text{on } H_0. \quad (1.52a)$$

To see that (1.50) implies (1.52a), note that

$$\langle A^*Ax, y \rangle \stackrel{(a)}{=} \langle Ax, Ay \rangle \stackrel{(b)}{=} \langle x, y \rangle,$$

where (a) follows from the definition of adjoint; and (b) from (1.50). Conversely, to see that (1.52a) implies (1.50), note that

$$\langle Ax, Ay \rangle \stackrel{(a)}{=} \langle x, A^*Ay \rangle \stackrel{(b)}{=} \langle x, y \rangle,$$

where (a) follows from the definition of adjoint; and (b) from (1.52a).

Combining (1.50) with invertibility gives that  $A^*$  is a right inverse of  $A$ :

$$AA^* = I \quad \text{on } H_1. \quad (1.52b)$$

To verify (1.52b), note that for every  $x, y$  in  $H_1$ ,

$$\langle AA^*x, y \rangle = \langle AA^*x, AA^{-1}y \rangle \stackrel{(a)}{=} \langle A^*x, A^{-1}y \rangle \stackrel{(b)}{=} \langle x, AA^{-1}y \rangle = \langle x, y \rangle,$$

where (a) follows from (1.50); and (b) from the definition of adjoint.

The desired equivalence follows: If  $A$  is unitary, then  $A$  is invertible and (1.50) holds, so both conditions (1.52) hold, so  $A^{-1} = A^*$ . Conversely, if  $A^{-1} = A^*$ , then  $A$  is invertible and (1.52a) holds, implying (1.50).

**Eigenvalues and Eigenvectors** The concept of an *eigenvector* generalizes from finite-dimensional linear algebra to our Hilbert space setting. Like other concepts that apply to matrices only when they are square, the generalization applies when the domain and codomain of a linear operator are the same Hilbert space. We call an eigenvector an *eigensequence* when the signal domain is  $\mathbb{Z}$  or a subset of  $\mathbb{Z}$  (for example, in Chapter 2); we call it an *eigenfunction* when the signal domain is  $\mathbb{R}$  or an interval  $[a, b]$  (for example, in Chapter 3).

**DEFINITION 1.24 (EIGENVECTOR OF A LINEAR OPERATOR)** An eigenvector of a linear operator  $A : H \rightarrow H$  is a nonzero vector  $v \in H$  such that

$$Av = \lambda v, \quad (1.53)$$

for some  $\lambda \in \mathbb{C}$ . The constant  $\lambda$  is called the corresponding eigenvalue and  $(\lambda, v)$  is called an eigenpair.

## 1.4. Approximations, Projections, and Decompositions

47

The eigenvalues and eigenvectors of a self-adjoint operator  $A$  have several useful properties:

- (i) All eigenvalues are real: If  $(\lambda, v)$  is an eigenpair,

$$\lambda \langle v, v \rangle = \langle \lambda v, v \rangle = \langle Av, v \rangle = \langle v, Av \rangle = \langle v, \lambda v \rangle = \lambda^* \langle v, v \rangle,$$

so  $\lambda$  is real.

- (ii) Eigenvectors corresponding to distinct eigenvalues are orthogonal: If  $(\lambda_0, v_0)$  and  $(\lambda_1, v_1)$  are eigenpairs with  $\lambda_0 \neq \lambda_1$ ,

$$\lambda_0 \langle v_0, v_1 \rangle = \langle \lambda_0 v_0, v_1 \rangle = \langle Av_0, v_1 \rangle = \langle v_0, Av_1 \rangle = \langle v_0, \lambda_1 v_1 \rangle = \lambda_1^* \langle v_0, v_1 \rangle,$$

so  $\langle v_0, v_1 \rangle = 0$ .

**Positive Definite Operators** Positive definiteness can also be generalized from square Hermitian matrices to self-adjoint operators on a general Hilbert space.

**DEFINITION 1.25 (DEFINITE LINEAR OPERATOR)** A self-adjoint operator  $A : H \rightarrow H$  is called

- (i) *positive semidefinite* or *nonnegative definite*, written  $A \geq 0$ , when

$$\langle Ax, x \rangle \geq 0 \quad \text{for all } x \in H; \quad (1.54a)$$

- (ii) *positive definite*, written  $A > 0$ , when

$$\langle Ax, x \rangle > 0 \quad \text{for all nonzero } x \in H; \quad (1.54b)$$

- (iii) *negative semidefinite* or *nonpositive definite* when  $-A$  is positive semidefinite; and

- (iv) *negative definite* when  $-A$  is positive definite.

As suggested by the notation, positive definiteness defines a partial order on self-adjoint operators. When  $A$  and  $B$  are self-adjoint operators defined on the same Hilbert space,  $A \geq B$  means  $A - B \geq 0$ , that is,  $A - B$  is a positive semidefinite operator.

As noted above, all eigenvalues of a self-adjoint operator are real. Positive definiteness is equivalent to all eigenvalues being positive; positive semidefiniteness is equivalent to all eigenvalues being nonnegative. Exercise 1.27 develops a proof of these facts.

## 1.4 Approximations, Projections, and Decompositions

Many of the linear operators that we encounter in later chapters are projection operators, in particular orthogonal projection operators. As we will see in this

section, orthogonal projection operators find best approximations from within a subspace, that is, approximations that minimize a Hilbert space norm of the error. An orthogonal projection generates a decomposition of a vector into components in two orthogonal subspaces. We will also see how the more general oblique projection operators generate decompositions of vectors into components in two subspaces that are not necessarily orthogonal.

### Best Approximation, Orthogonal Projection, and Orthogonal Decomposition

Let  $S$  be a closed subspace of a Hilbert space  $H$  and let  $x$  be a vector in  $H$ . The best approximation problem is to find the vector in  $S$  that is closest to  $x$ :

$$\hat{x} = \arg \min_{s \in S} \|x - s\|. \quad (1.55)$$

Most commonly, a Hilbert space norm involves a sum of squares (as in  $\ell^2(\mathbb{Z})$ ) or integral of a square (as in  $\mathcal{L}^2(\mathbb{R})$ ), in which case  $\hat{x}$  is called a *least-squares approximation*. Of course, when  $x$  is in  $S$  then  $\hat{x} = x$  uniquely solves the problem—it makes the approximation error  $\|x - \hat{x}\|$  zero.<sup>11</sup> The interesting case is when  $x$  is not in  $S$ .

Figure 1.11 illustrates the problem and its solution in ordinary Euclidean geometry. The point on the line  $S$  that is closest to a point  $x$  not on the line is uniquely determined by finding the circle centered at  $x$  that is tangent to the line. Any other candidate  $x'$  on the line lies outside the circle and is thus farther from  $x$ . Since a line tangent to a circle is always perpendicular to a line segment from the tangent point to the center of the circle, the solution  $\hat{x}$  satisfies the following orthogonality property:  $x - \hat{x} \perp S$ . The projection theorem extends this geometric result to arbitrary Hilbert spaces. The approximation  $\hat{x}$  and the residual  $x - \hat{x}$  form an orthogonal decomposition of  $x$  that is uniquely determined by the subspace  $S$ , and orthogonality of  $x - \hat{x}$  and  $S$  uniquely determines  $\hat{x}$ .

#### 1.4.1 Projection Theorem

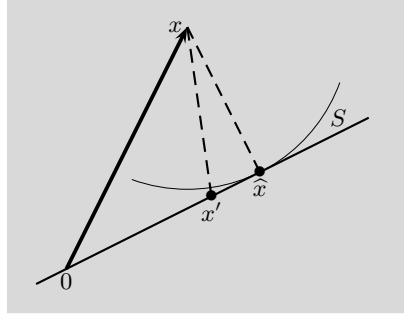
The solution to the best approximation problem in a general Hilbert space is described by the following theorem:

**THEOREM 1.26 (PROJECTION THEOREM)** Let  $S$  be a closed subspace of Hilbert space  $H$  and let  $x$  be a vector in  $H$ .

- (i) *Existence:* There exists  $\hat{x} \in S$  such that  $\|x - \hat{x}\| \leq \|x - s\|$  for all  $s \in S$ .
- (ii) *Orthogonality:*  $x - \hat{x} \perp S$  is necessary and sufficient for determining  $\hat{x}$ .
- (iii) *Uniqueness:* The vector  $\hat{x}$  is unique.
- (iv) *Linearity:*  $\hat{x} = Px$  where  $P$  is a linear operator that depends on  $S$  and not on  $x$ .

<sup>11</sup>Recall from Definition 1.9(i) that  $\|x - \hat{x}\|$  is nonnegative and zero if and only if  $x - \hat{x} = 0$ .





**Figure 1.11:** Illustration of best approximation. In Euclidean geometry, the best approximation of  $x$  on the line  $S$  is obtained with error  $x - \hat{x}$  orthogonal to  $S$ ; any candidate  $x'$  such that  $x - x'$  is not orthogonal to  $S$  is farther from  $x$ . This holds more generally in Hilbert spaces.

- (v) *Idempotency:*  $P(Px) = Px$  for all  $x \in H$ .
- (vi) *Self-adjointness:*  $P = P^*$ .

*Proof.* We prove existence last since it is the most technical and is the only part that requires completeness of the space. (Orthogonality and uniqueness hold with  $H$  replaced by any inner product space and  $S$  replaced by any subspace.)

- (ii) *Orthogonality:* Suppose that  $\hat{x}$  minimizes  $\|x - \hat{x}\|$  but that  $x - \hat{x} \not\perp S$ . Then there exists a unit vector  $\varphi \in S$  such that  $\langle x - \hat{x}, \varphi \rangle = \varepsilon \neq 0$ . Let  $s = \hat{x} + \varepsilon\varphi$  and note that  $s$  is in  $S$  since  $S$  is a subspace. The calculation

$$\begin{aligned} \|x - s\|^2 &= \|x - \hat{x} - \varepsilon\varphi\|^2 \\ &= \|x - \hat{x}\|^2 - \underbrace{\langle x - \hat{x}, \varepsilon\varphi \rangle}_{= |\varepsilon|^2} - \underbrace{\langle \varepsilon\varphi, x - \hat{x} \rangle}_{= |\varepsilon|^2} + \underbrace{\|\varepsilon\varphi\|^2}_{= |\varepsilon|^2} \\ &= \|x - \hat{x}\|^2 - |\varepsilon|^2 < \|x - \hat{x}\|^2 \end{aligned}$$

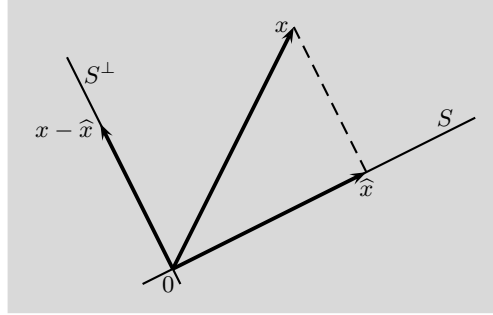
then shows that  $\hat{x}$  is not the minimizing vector. This contradiction implies  $x - \hat{x} \perp S$ . This can also be written as  $x - \hat{x} \in S^\perp$ ; see Figure 1.12.

- (iii) *Uniqueness:* Suppose  $x - \hat{x} \perp S$ . For any  $s \in S$ ,

$$\|x - s\|^2 = \|(x - \hat{x}) + (\hat{x} - s)\|^2 \stackrel{(a)}{=} \|x - \hat{x}\|^2 + \|\hat{x} - s\|^2,$$

where  $\hat{x} - s \in S$  implies  $x - \hat{x} \perp \hat{x} - s \in S$ , which allows an application of the Pythagorean theorem in (a). This shows  $\|x - s\| > \|x - \hat{x}\|$  for any  $s \neq \hat{x}$ .

- (iv) *Linearity:* Let  $\alpha$  be any scalar, and denote the best approximations in  $S$  of  $x_1$  and  $x_2$  by  $\hat{x}_1$  and  $\hat{x}_2$ . The orthogonality property implies  $x_1 - \hat{x}_1 \in S^\perp$  and  $x_2 - \hat{x}_2 \in S^\perp$ . Since  $S$  is a subspace,  $\hat{x}_1 + \hat{x}_2 \in S$ ; and since  $S^\perp$  is a subspace,  $(x_1 - \hat{x}_1) + (x_2 - \hat{x}_2) \in S^\perp$ . With  $(x_1 + x_2) - (\hat{x}_1 + \hat{x}_2) \in S^\perp$  and  $\hat{x}_1 + \hat{x}_2 \in S$ , the uniqueness property shows that  $\hat{x}_1 + \hat{x}_2$  is the best approximation of  $x_1 + x_2$ .



**Figure 1.12:** The best approximation of  $x \in H$  within closed subspace  $S$  is uniquely determined by  $x - \hat{x} \perp S$ . The solution generates an orthogonal decomposition of  $x$  into  $\hat{x} \in S$  and  $x - \hat{x} \in S^\perp$ .

This shows additivity. Similarly, since  $S$  is a subspace,  $\alpha\hat{x}_1 \in S$ ; and since  $S^\perp$  is a subspace,  $\alpha(x_1 - \hat{x}_1) \in S^\perp$ . With  $\alpha x_1 - \alpha\hat{x}_1 \in S^\perp$  and  $\alpha\hat{x}_1 \in S$ , the uniqueness property shows that  $\alpha\hat{x}_1$  is the best approximation of  $\alpha x_1$ . This shows scalability.

- (v) *Idempotency:* The property to check is that the operator  $P$  leaves  $Px$  unchanged. This follows from two facts:  $Px \in S$  and  $Pu = u$  for all  $u \in S$ . That  $Px$  is in  $S$  is part of the definition of  $\hat{x}$ . For the second fact, let  $u \in S$  and suppose  $\hat{u}$  satisfies

$$\|u - \hat{u}\| \leq \|u - s\| \quad \text{for all } s \in S.$$

By the uniqueness property, there can be only one such  $\hat{u}$ , and since  $\hat{u} = u$  gives  $\|u - \hat{u}\| = 0$  and the norm is nonnegative, we must have  $\hat{u} = u$ .

- (vi) *Self-adjointness:* We would like to show  $\langle Px, y \rangle = \langle x, Py \rangle$  for all  $x, y \in H$ :

$$\langle Px, y \rangle = \langle Px, Py + (y - Py) \rangle \stackrel{(a)}{=} \langle Px, Py \rangle + \langle Px, y - Py \rangle \stackrel{(b)}{=} \langle Px, Py \rangle,$$

where (a) uses the distributivity of the inner product; and (b) follows from  $Px \in S$  and  $y - Py \in S^\perp$ ; similarly

$$\langle x, Py \rangle = \langle Px + (x - Px), Py \rangle = \langle Px, Py \rangle + \langle x - Px, Py \rangle = \langle Px, Py \rangle.$$

- (i) *Existence:* We finally show existence of a minimizing  $\hat{x}$ . If  $x$  is in  $S$ , then  $\hat{x} = x$  achieves the minimum so there is no question of existence. We thus restrict our attention to  $x \notin S$ . Let  $\varepsilon = \inf_{s \in S} \|x - s\|$ . Then there exists a sequence of vectors  $s_0, s_1, \dots$  in  $S$  such that  $\|x - s_k\| \rightarrow \varepsilon$ ; the challenge is to show that the infimum is achieved by some  $\hat{x} \in S$ . We do this by showing that  $\{s_k\}_{k \geq 0}$  is a Cauchy sequence and thus converges, within the closed subspace  $S$ , to the desired  $\hat{x}$ .

By applying the parallelogram law (1.25) to  $x - s_j$  and  $s_i - x$ ,

$$\|(x - s_j) + (s_i - x)\|^2 + \|(x - s_j) - (s_i - x)\|^2 = 2\|x - s_j\|^2 + 2\|s_i - x\|^2.$$

Cancelling  $x$  in the first term and moving the second term to the right yields

$$\|s_i - s_j\|^2 = 2\|x - s_j\|^2 + 2\|s_i - x\|^2 - 4\|x - \frac{1}{2}(s_i + s_j)\|^2. \quad (1.56)$$

## 1.4. Approximations, Projections, and Decompositions

51

Now since  $S$  is a subspace,  $\frac{1}{2}(s_i + s_j)$  is in  $S$ . Thus by the definition of  $\varepsilon$  we have  $\|x - \frac{1}{2}(s_i + s_j)\| \geq \varepsilon$ . Substituting in (1.56) and using the nonnegativity of the norm,

$$0 \leq \|s_i - s_j\|^2 \leq 2\|x - s_j\|^2 + 2\|s_i - x\|^2 - 4\varepsilon^2.$$

With the convergence of  $\|x - s_j\|^2$  and  $\|s_i - x\|^2$  to  $\varepsilon^2$ , we conclude that  $\{s_k\}_{k \geq 0}$  is a Cauchy sequence. Now since  $S$  is a closed subspace of a complete space,  $\{s_k\}_{k \geq 0}$  converges to  $\hat{x} \in S$ . Since a norm is continuous, convergence of  $\{s_k\}_{k \geq 0}$  to  $\hat{x}$  implies  $\|x - \hat{x}\| = \varepsilon$ .

The projection theorem leads to a simple and unified methodology for computing best approximations through the normal equations. We develop this in detail after the introduction of bases in Section 1.5. The following example provides a preview.

**EXAMPLE 1.17** Consider the function  $x(t) = \cos(\frac{3}{2}\pi t)$  in the Hilbert space  $\mathcal{L}^2([0, 1])$ . To find the degree-1 polynomial closest to  $x$  directly (without using the projection theorem) would require solving

$$\min_{a_0, a_1} \int_0^1 |\cos(\frac{3}{2}\pi t) - (a_0 + a_1 t)|^2 dt.$$

Noting that the degree-1 polynomials form a closed subspace in this Hilbert space, the projection theorem shows that  $(a_0, a_1)$  is determined uniquely by requiring

$$x(t) - \hat{x}(t) = \cos(\frac{3}{2}\pi t) - (a_0 + a_1 t)$$

to be orthogonal to the entire subspace of degree-1 polynomials. Imposing orthogonality to 1 and  $t$  gives two linearly-independent equations to solve:

$$\begin{aligned} 0 &= \langle x(t) - \hat{x}(t), 1 \rangle = \int_0^1 (\cos(\frac{3}{2}\pi t) - (a_0 + a_1 t)) \cdot 1 dt = -\frac{2}{3\pi} - a_0 - \frac{1}{2}a_1, \\ 0 &= \langle x(t) - \hat{x}(t), t \rangle = \int_0^1 (\cos(\frac{3}{2}\pi t) - (a_0 + a_1 t)) \cdot t dt = \frac{4 + 6\pi}{9\pi^2} - \frac{1}{2}a_0 - \frac{1}{3}a_1. \end{aligned}$$

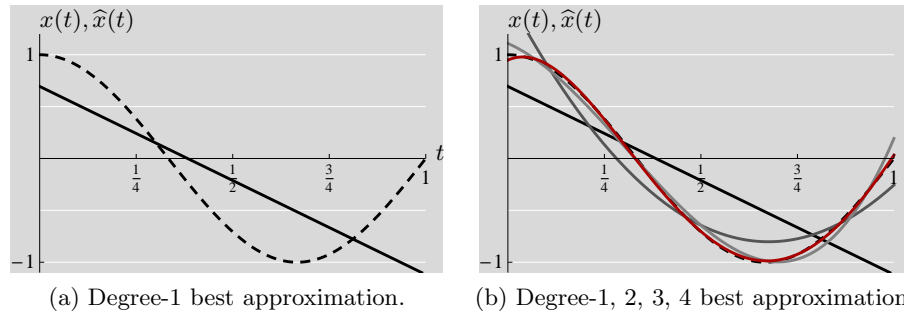
Their solution is

$$a_0 = \frac{8 + 4\pi}{3\pi^2}, \quad a_1 = -\frac{16 + 12\pi}{3\pi^2}.$$

Figure 1.13(a) shows the function and its degree-1 polynomial approximation.

Best approximations among degree- $k$  polynomials for  $k = 1, 2, 3, 4, 5$  are shown in Figure 1.13(b). Increasing the degree increases the size of the subspace of polynomials, so the quality of approximation is naturally improved. We will see in Chapter 5 (Theorem 5.3) that since  $x(t)$  is continuous, the error is driven to zero by letting  $k$  grow without bound.

The effect of the operator  $P$  that arises in the projection theorem is to move the input vector  $x$  in a direction orthogonal to the subspace  $S$  until  $S$  is reached at  $\hat{x}$ . In the following two sections, we show that  $P$  has the defining characteristics of what we will call an orthogonal projection operator, and we describe the mapping of  $x$  to an approximation  $\hat{x}$  and residual  $x - \hat{x}$  as what we will call an orthogonal decomposition. Projections and decompositions each have nonorthogonal (oblique) versions.



**Figure 1.13:** (a) The best approximation of  $x(t)$  (dashed) among degree-1 polynomials is  $\hat{x}(t)$  (solid), where approximation quality is measured by the  $\mathcal{L}^2$  norm on  $([0, 1])$ ; see Example 1.17. The best approximation is determined by the orthogonality of  $x - \hat{x}$  to the subspace of degree-1 polynomials. (b) Allowing higher-degree polynomials, for  $k = 1$  (black),  $k = 2$  (dark gray),  $k = 3$  (light gray),  $k = 4$  (dark red), increases the size of the subspace to which  $x(t)$  is orthogonally projected and decreases the approximation error.

## 1.4.2 Projection Operators

The operator  $P$  that arises from solving the best approximation problem is an orthogonal projection operator, as per the following definition:<sup>12</sup>

**DEFINITION 1.27 (PROJECTION, ORTHOGONAL PROJECTION, OBLIQUE PROJECTION)**

- (i) An idempotent operator  $P$  is an operator such that  $P^2 = P$ .
- (ii) A projection operator is a bounded linear operator that is idempotent.
- (iii) An orthogonal projection operator is a projection operator that is self-adjoint.
- (iv) An oblique projection operator is a projection operator that is not self-adjoint.

An operator is idempotent when applying it twice is no different than applying it once. Setting certain components of a vector to a constant value is an idempotent operation, and when that constant is zero this operation is linear. The following example introduces a notation for the basic class of orthogonal projection operators that set a portion of a vector to zero.

<sup>12</sup>Two notes of caution are warranted. First, we use the term projection only with respect to Hilbert space geometry; many other mathematical and scientific meanings are inconsistent with this. Second, some authors use “projection” to mean “orthogonal projection.” We will *not* adopt this potentially-confusing shorthand because many important properties and uses of projection operators hold for all projections—both orthogonal and oblique.

## 1.4. Approximations, Projections, and Decompositions

53

EXAMPLE 1.18 (PROJECTION VIA DOMAIN RESTRICTION) Let  $\mathcal{I}$  be a subset of  $\mathbb{Z}$ , and define the linear operator  $1_{\mathcal{I}} : \ell^2(\mathbb{Z}) \rightarrow \ell^2(\mathbb{Z})$  by

$$y = 1_{\mathcal{I}} x \quad \text{where} \quad y_k = \begin{cases} x_k, & \text{for } k \in \mathcal{I}; \\ 0, & \text{otherwise.} \end{cases} \quad (1.57)$$

This is a special case of the linear operator in Example 1.14(iii), with

$$\alpha_k = \begin{cases} 1, & \text{for } k \in \mathcal{I}; \\ 0, & \text{otherwise.} \end{cases}$$

This operator is obviously idempotent, and it is self-adjoint because of the adjoint computation in Example 1.15(iii). Thus  $1_{\mathcal{I}}$  is an orthogonal projection operator.

The same notation is used for vector spaces with domains other than  $\mathbb{Z}$ . For example, with  $\mathcal{I}$  a subset of  $\mathbb{R}$ , we define the linear operator  $1_{\mathcal{I}} : \mathcal{L}^2(\mathbb{R}) \rightarrow \mathcal{L}^2(\mathbb{R})$  by

$$y = 1_{\mathcal{I}} x \quad \text{where} \quad y(t) = \begin{cases} x(t), & \text{for } t \in \mathcal{I}; \\ 0, & \text{otherwise.} \end{cases} \quad (1.58)$$

Exercise 1.30 establishes properties of this operator.

The following theorem uses orthogonality of certain vectors to prove that an operator is an orthogonal projection operator. This complements the projection theorem, since here an operator is specified rather than a subspace. We discuss the subspace that is implicit in the theorem after the proof.

THEOREM 1.28 A bounded linear operator  $P$  on a Hilbert space  $H$  satisfies

$$\langle x - Px, Py \rangle = 0 \quad \text{for all } x, y \in H \quad (1.59)$$

if and only if  $P$  is an orthogonal projection operator.

*Proof.* We first prove that idempotency and self-adjointness of  $P$  imply that (1.59) holds. With  $x$  and  $y$  arbitrary vectors in  $H$ ,

$$\begin{aligned} \langle x - Px, Py \rangle &= \langle x, Py \rangle - \langle Px, Py \rangle \stackrel{(a)}{=} \langle x, Py \rangle - \langle x, P^* Py \rangle \\ &\stackrel{(b)}{=} \langle x, Py \rangle - \langle x, P^2 y \rangle \stackrel{(c)}{=} \langle x, Py \rangle - \langle x, Py \rangle = 0, \end{aligned}$$

where (a) follows from the definition of the adjoint operator; (b) from self-adjointness of  $P$ ; and (c) from idempotency of  $P$ .

Now suppose that (1.59) holds. With  $z \in H$  arbitrary, since (1.59) holds for  $x = Pz$  and  $y = z - Pz$ , we get

$$0 = \langle Pz - P(Pz), P(z - Pz) \rangle = \|(P - P^2)z\|^2,$$

which implies  $Pz = P^2 z$ . Since  $z$  is arbitrary, we have  $P = P^2$  (idempotency of  $P$ ).

By Hermitian symmetry of the inner product, (1.59) implies

$$\langle Px, y - Py \rangle = 0 \quad \text{for all } x, y \in H. \quad (1.60)$$

Thus, for any  $x, y \in H$ ,

$$\langle P^*x, y \rangle \stackrel{(a)}{=} \langle x, Py \rangle \stackrel{(b)}{=} \langle Px, Py \rangle \stackrel{(c)}{=} \langle Px, y \rangle,$$

where (a) uses the definition of adjoint; (b) uses (1.59); and (c) uses (1.60). This implies  $\langle P^*x - Px, y \rangle = 0$ . By choosing  $y = P^*x - Px$ , we have that  $\|P^*x - Px\| = 0$  for all  $x$ , and thus  $P^* = P$  (self-adjointness of  $P$ ).

The range of any linear operator is a subspace. In the setting of the preceding theorem, we may associate with  $P$  the closed subspace  $S = \mathcal{R}(P)$ . Then we have that  $P$  is the orthogonal projection operator onto  $S$ . The orthogonality equation (1.59) is a restatement of the projection residual  $x - Px$  being orthogonal to  $S$ .

The following example gives an explicit expression for projection operators onto 1-dimensional subspaces. This was discussed informally in Section 1.1.

**EXAMPLE 1.19 (ORTHOGONAL PROJECTION ONTO 1-DIMENSIONAL SUBSPACE)**  
Given a vector  $\varphi \in H$  of unit norm, let

$$Px = \langle x, \varphi \rangle \varphi. \quad (1.61)$$

This is a linear operator because of the distributivity and linearity in the first argument of the inner product. To use Theorem 1.28 to show that  $P$  is the orthogonal projection operator onto the subspace of scalar multiples of  $\varphi$ , we verify the idempotency and self-adjointness of  $P$ . Idempotency is proven by the following computation:

$$P^2x = \langle \langle x, \varphi \rangle \varphi, \varphi \rangle \varphi \stackrel{(a)}{=} \langle x, \varphi \rangle \langle \varphi, \varphi \rangle \varphi \stackrel{(b)}{=} \langle x, \varphi \rangle \varphi = Px,$$

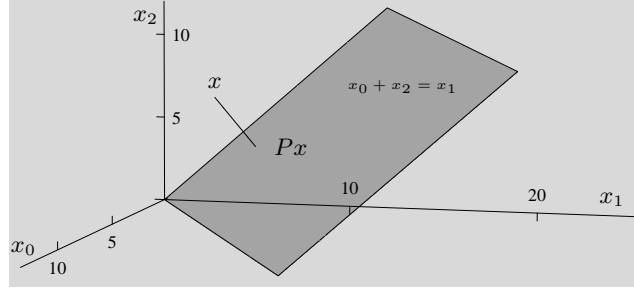
where (a) uses the linearity in the first argument of the inner product; and (b) follows from  $\langle \varphi, \varphi \rangle = 1$ . Self-adjointness is proven by the following computation:

$$\begin{aligned} \langle Px, y \rangle &= \langle \langle x, \varphi \rangle \varphi, y \rangle \stackrel{(a)}{=} \langle x, \varphi \rangle \langle \varphi, y \rangle = \langle \varphi, y \rangle \langle x, \varphi \rangle \\ &\stackrel{(b)}{=} \langle y, \varphi \rangle^* \langle x, \varphi \rangle \stackrel{(c)}{=} \langle x, \langle y, \varphi \rangle \varphi \rangle = \langle x, Py \rangle, \end{aligned}$$

where (a) follows from linearity in the first argument of the inner product; (b) from conjugate symmetry of the inner product; and (c) from conjugate linearity in the second argument of the inner product.

Collections of 1-dimensional projections are central to representations using bases, which are introduced in Section 1.5 and developed in several subsequent chapters. Solved Exercise 1.4 extends the 1-dimensional example to orthogonal projection onto subspaces of higher dimensions.

The final theorem of the section establishes important connections between inverses, adjoints and projections; its simple proof is left for Exercise 1.31.



**Figure 1.14:** Two-dimensional range of the oblique projection operator  $P$  from Example 1.20. It is the plane  $x_0 + x_2 = x_1$ . For example, vector  $x = [6 \ 6 \ 8]^T$  is projected via  $P$  onto  $Px = [2 \ 6 \ 4]^T$ , not an orthogonal projection.

**THEOREM 1.29** Let  $A : H_0 \rightarrow H_1$  and  $B : H_1 \rightarrow H_0$  be bounded linear operators. If  $A$  is a left inverse of  $B$ , then  $BA$  is a projection operator. Furthermore, if  $B = A^*$ , then  $BA = A^*A$  is an orthogonal projection operator.

**EXAMPLE 1.20 (PROJECTION ONTO A SUBSPACE)** Let

$$A = \frac{1}{2} \begin{bmatrix} 1 & 1 & -1 \\ -1 & 1 & 1 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}.$$

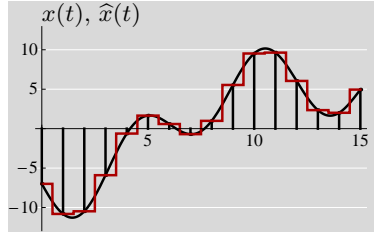
Since  $A$  is a left inverse of  $B$ , we know from Theorem 1.29 that  $P = BA$  is a projection operator. Explicitly,

$$P = BA = \frac{1}{2} \begin{bmatrix} 1 & 1 & -1 \\ 0 & 2 & 0 \\ -1 & 1 & 1 \end{bmatrix},$$

from which one can verify  $P^2 = P$ . A description of the 2-dimensional range of this projection operator is most transparent from  $B$ : it is the set of 3-tuples with middle component equal to the sum of the first and last (see Figure 1.14). Note that  $P \neq P^*$ , so the projection is oblique.

The final example of the section draws together some earlier results to give an orthogonal projection operator on  $\mathcal{L}^2(\mathbb{R})$ . It illustrates the basics of sampling and interpolation, to which we will return in Chapter 4.

**EXAMPLE 1.21 (ORTHOGONAL PROJECTION OPERATOR ON  $\mathcal{L}^2(\mathbb{R})$ )** Let  $A : \mathcal{L}^2(\mathbb{R}) \rightarrow \ell^2(\mathbb{Z})$  be the local averaging operator (1.45) and let  $A^* : \ell^2(\mathbb{Z}) \rightarrow \mathcal{L}^2(\mathbb{R})$  be its adjoint, as derived in Example 1.16. If we verify that  $A$  is a left inverse of  $A^*$ , we will have as a consequence of Theorem 1.29 that  $A^*A$  is an orthogonal projection operator.



**Figure 1.15:** Illustration of an orthogonal projection operator on  $\mathcal{L}^2(\mathbb{R})$ . The linear operator  $A$  and its adjoint  $A^*$  illustrated in Figure 1.10 satisfy  $AA^* = I$ , so  $A^*A$  is an orthogonal projection operator. The range of  $A^*$  is the subspace of  $\mathcal{L}^2(\mathbb{R})$  consisting of functions that are constant on all intervals  $[n - 1/2, n + 1/2)$ ,  $n \in \mathbb{Z}$ . Thus,  $\hat{x} = A^*Ax$  (piecewise constant, red) is the best approximation of  $x$  (smooth, blue) in this subspace.

To check that  $A$  is a left inverse of  $A^*$ , consider the application of  $AA^*$  to an arbitrary sequence in  $\ell^2(\mathbb{Z})$ . (Recall that the separate effects of  $A$  and  $A^*$  are illustrated in Figure 1.10.) Remembering to compose from right to left,  $AA^*$  starts with a sequence  $x_n$ , creates a function equal to  $x_n$  on each interval  $[n - \frac{1}{2}, n + \frac{1}{2})$ , and then recovers the original sequence by finding the average value of the function on each interval  $[n - \frac{1}{2}, n + \frac{1}{2})$ . So  $AA^*$  is indeed an identity operator. One conclusion to draw by combining the projection theorem with  $A^*A$  being an orthogonal projection operator is the following: Given a function  $x(t) \in \mathcal{L}^2(\mathbb{R})$ , the function in the subspace of piecewise constant functions  $A^*\ell^2(\mathbb{Z})$  that is closest to  $x(t)$  in  $\mathcal{L}^2$  norm is the one obtained by replacing  $x(t)$ ,  $t \in [n - \frac{1}{2}, n + \frac{1}{2})$ , by its local average  $\int_{n-1/2}^{n+1/2} x(t) dt$ . The result of applying  $A^*A$  is depicted in Figure 1.15.

### 1.4.3 Direct Sums and Subspace Decompositions

In the projection theorem, the best approximation  $\hat{x}$  is uniquely determined by the orthogonality of  $\hat{x}$  and  $x - \hat{x}$ . Thus, the projection theorem proves that  $x$  can be written uniquely as

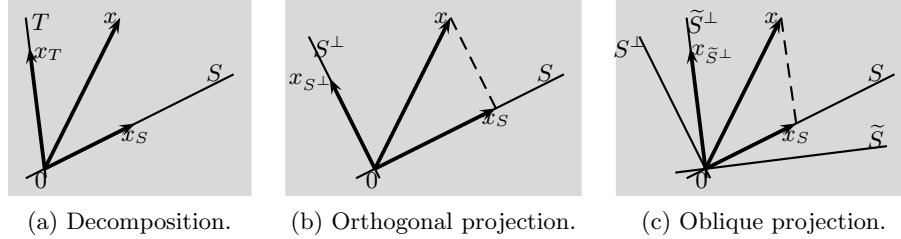
$$x = x_S + x_{S^\perp} \quad \text{where } x_S \in S \text{ and } x_{S^\perp} \in S^\perp, \quad (1.62)$$

because we uniquely choose  $x_S = \hat{x}$  and  $x_{S^\perp} = x - \hat{x}$ . Being able to uniquely write the sum (1.62) for any  $x$  defines a decomposition; while having an orthogonal pair of subspaces is an important case, it is not necessary.

**DEFINITION 1.30 (DIRECT SUM AND DECOMPOSITION)** A vector space  $V$  is a direct sum of subspaces  $S$  and  $T$ , denoted  $V = S \oplus T$ , when any nonzero vector  $x \in V$  can be written uniquely as

$$x = x_S + x_T \quad \text{where } x_S \in S \text{ and } x_T \in T. \quad (1.63)$$





**Figure 1.16:** (a) A vector space  $V$  is decomposed as a direct sum  $S \oplus T$  when any  $x \in V$  can be written uniquely as a sum of components in  $S$  and  $T$ . (b) An orthogonal projection operator generates an orthogonal direct sum decomposition of a Hilbert space. It decomposes vector  $x$  into  $x_S \in S$  and  $x_{S^\perp} \in S^\perp$ . (c) An oblique projection operator generates a nonorthogonal direct sum decomposition of a Hilbert space. It decomposes vector  $x$  into  $x_S \in S$  and  $x_{\tilde{S}^\perp} \in \tilde{S}^\perp$ .

The subspaces  $S$  and  $T$  form a decomposition of  $V$ , and the vectors  $x_S$  and  $x_T$  form a decomposition of  $x$ . When  $S$  and  $T$  are orthogonal, this is called an orthogonal decomposition.

A general direct sum decomposition  $V = S \oplus T$  is illustrated in Figure 1.16(a).

When  $S$  is a closed subspace of a Hilbert space  $H$ , the projection theorem generates the unique decomposition (1.62); thus,  $H = S \oplus S^\perp$ . It is tempting to write  $V = S \oplus S^\perp$  for any (not necessarily closed) subspace of any (not necessarily complete) vector space. However, this is not always possible. The following example highlights the necessity of working with closed subspaces of a complete space. As noted before Example 1.11, we frequently work the closure of a span to avoid the pitfalls of subspaces that are not closed.

**EXAMPLE 1.22 (FAILURE OF DIRECT SUM  $S \oplus S^\perp$ )** As in Example 1.11, consider Hilbert space  $\ell^2(\mathbb{N})$  and for each  $k \in \mathbb{N}$ , let the sequence  $s_k$  be 0 except for a 1 in the  $k$ th position. Then  $S = \text{span}(\{s_0, s_1, \dots\})$  consists of all vectors in  $\ell^2(\mathbb{N})$  that have a finite number of nonzero entries, and  $S$  is a subspace. Let  $x \in \ell^2(\mathbb{N})$  be a nonzero vector. Then  $x_n \neq 0$  for some  $n \in \mathbb{N}$ , so  $x \notin S$ , which implies  $x \notin S^\perp$ . Since no nonzero vector is in  $S^\perp$ , we have that  $S^\perp = \{\mathbf{0}\}$ . Since  $S$  itself is not all of  $\ell^2(\mathbb{N})$ , one cannot write every  $x \in \ell^2(\mathbb{N})$  as in (1.62).

The main aim of this section is to extend the connection between decompositions and projections to the general (oblique) case. The following theorem establishes that a projection operator  $P$  generates a direct sum decomposition as illustrated in Figure 1.16(c). The dashed line shows the effect of the operator,  $x_S = Px$ , and the residual  $x_{\tilde{S}^\perp} = x - x_S$  is in a subspace we denote  $\tilde{S}^\perp$  rather than  $T$  for reasons that will become clear.

**THEOREM 1.31** Let  $H$  be a Hilbert space.

- (i) Let  $P$  be a projection operator on  $H$ , and let  $S = \mathcal{R}(P)$  and  $T = \mathcal{N}(P)$ . Then  $H = S \oplus T$ .
- (ii) Conversely, let closed subspaces  $S$  and  $T$  satisfy  $H = S \oplus T$ . Then there exists a projection operator on  $H$  such that  $S = \mathcal{R}(P)$  and  $T = \mathcal{N}(P)$ .

*Proof.* (i) Let  $x \in H$ . We would like to prove that a decomposition of the form (1.63) exists and is unique. Existence is verified by letting  $x_S = Px$ , which obviously is in  $S = \mathcal{R}(P)$ ; and  $x_T = x - Px$ , which is in  $T = \mathcal{N}(P)$  because

$$Px_T = P(x - Px) = Px - P^2x \stackrel{(a)}{=} Px - Px = \mathbf{0},$$

where (a) uses that  $P$  is idempotent. For uniqueness, suppose

$$x = x'_S + x'_T \quad \text{where } x'_S \in S \text{ and } x'_T \in T.$$

Equating the two expansions of  $x$  and applying  $P$ , we have

$$\begin{aligned} \mathbf{0} &= P((x_S - x'_S) + (x_T - x'_T)) = P(x_S - x'_S) + P(x_T - x'_T) \\ &\stackrel{(a)}{=} P(x_S - x'_S) \stackrel{(b)}{=} x_S - x'_S, \end{aligned}$$

where (a) follows from  $x_T - x'_T$  lying in  $T$ , the null space of  $P$ ; and (b) follows from  $x_S - x'_S$  lying in  $S$  and  $P$  equaling the identity on  $S$ . From this,  $x'_S = x_S$  and  $x'_T = x_T$  follow.

- (ii) Define the desired projection operator  $P$  from the unique decomposition of any  $x \in H$  of the form (1.63) through  $Px = x_S$ . The linearity of  $P$  follows easily from the assumed uniqueness of decompositions of vectors. By construction, the range of  $P$  is contained in  $S$ . It is actually all of  $S$  because any  $x \in S$  can be uniquely decomposed as  $x + \mathbf{0}$  with  $x \in S$  and  $\mathbf{0} \in T$ . Similarly, the null space of  $P$  contains  $T$  because any  $x \in T$  can be uniquely decomposed as  $\mathbf{0} + x$  with  $\mathbf{0} \in S$  and  $x \in T$ , showing that  $Px = \mathbf{0}$ . The null space of  $P$  is not larger than  $T$  because any vector  $x \in H \setminus T$  can be written uniquely as in (1.63) with  $x_S \neq \mathbf{0}$ , so  $Px \neq \mathbf{0}$ . It remains only to verify that  $P$  is idempotent. This follows from  $Px \in S$  and that  $P$  equals the identity on  $S$ .

The following example makes explicit the form of a (possibly oblique) projection when  $S$  is a 1-dimensional subspace. For consistency with later developments, the illustration of Theorem 1.31 in Figure 1.16(c) uses  $\tilde{S}^\perp$  in place of  $T$ . Since  $S$  and  $T = \tilde{S}^\perp$  have complementary dimension (adding to the whole Hilbert space), we have that  $S$  and  $\tilde{S}$  are of the same dimension. When  $S = \tilde{S}$ , the projection and decomposition are orthogonal, and Figure 1.16(c) reduces to Figure 1.12. In the example, this corresponds to  $\varphi = \tilde{\varphi}$ .

**EXAMPLE 1.23 (OBLIQUE PROJECTION ONTO 1-DIMENSIONAL SUBSPACE)** Let  $S$  be the multiples of vector  $\varphi \in H$  of unit norm, and let  $\tilde{S}$  be the multiples of an arbitrary vector  $\tilde{\varphi} \in H$ . The operator

$$Px = \langle x, \tilde{\varphi} \rangle \varphi \tag{1.64}$$

is linear and has range contained in  $S$ . We will find conditions under which it generates a decomposition  $H = S \oplus \tilde{S}^\perp$ .

Since

$$P^2x = \langle \langle x, \tilde{\varphi} \rangle \varphi, \tilde{\varphi} \rangle \varphi \stackrel{(a)}{=} \langle x, \tilde{\varphi} \rangle \langle \varphi, \tilde{\varphi} \rangle \varphi,$$

where (a) uses the linearity in the first argument of the inner product,  $P$  is idempotent if and only if  $\langle \varphi, \tilde{\varphi} \rangle = 1$ . Under this condition, we have  $H = \mathcal{R}(P) \oplus \mathcal{N}(P)$  by Theorem 1.31. This can also be written as  $H = S \oplus \tilde{S}^\perp$  because  $\mathcal{N}(P)$  and  $\tilde{S}^\perp$  are both precisely the set  $\{x \in H \mid \langle x, \tilde{\varphi} \rangle = 0\}$ .

If  $\langle \varphi, \tilde{\varphi} \rangle \neq 1$  but also  $\langle \varphi, \tilde{\varphi} \rangle \neq 0$ , a simple adjustment of the length of  $\tilde{\varphi}$  will make  $P$  a projection operator. However, if  $\langle \varphi, \tilde{\varphi} \rangle = 0$ , it is not possible to decompose  $H$  as desired. In fact,  $S$  and  $\tilde{S}$  are orthogonal, so  $S$  and  $\tilde{S}^\perp$  are the same subspace.

#### 1.4.4 Minimum Mean-Squared Error Estimation

The set of complex random variables can be viewed as a vector space over the complex numbers (see Section 1.2.4). With the restriction of finite second moments—which is implicit through the remainder of this section—this set forms a Hilbert space under the inner product (1.33):

$$\langle x, y \rangle = E[xy^*].$$

The square of the norm of the difference between vectors in this vector space is the *mean-squared error* (MSE) between the random variables:

$$\|x - \hat{x}\|^2 = E[|x - \hat{x}|^2]. \quad (1.65)$$

Since minimizing MSE is equivalent to minimizing a Hilbert space norm, many minimum MSE (MMSE) estimation problems are solved easily with the projection theorem. Throughout this section, MMSE estimators are called *optimal*, whether or not the estimator is constrained to a particular form.

**Linear Estimation** Let  $x$  and  $y_1, y_2, \dots, y_K$  be jointly-distributed complex random variables. A linear estimator<sup>13</sup> of  $x$  from the  $y_k$ s is a random variable of the form

$$\hat{x} = \alpha_0 + \alpha_1 y_1 + \alpha_2 y_2 + \dots + \alpha_K y_K. \quad (1.66)$$

When the coefficients are chosen to minimize the MSE, the estimator is called the *linear minimum mean-squared error* (LMMSE) estimator. Since (1.66) places  $\hat{x}$  in a closed subspace  $S$  of a Hilbert space of random variables, the projection theorem dictates that the optimal estimator  $\hat{x}$  must be such that the error  $x - \hat{x}$  is orthogonal to the subspace:  $x - \hat{x} \perp S$ .

<sup>13</sup>A function of the form  $f(x_1, x_2, \dots, x_K) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \dots + \alpha_K x_K$ , where the  $\alpha_k$ s are constants, is called *affine*. The estimator in (1.66) is called linear even though  $\hat{x}$  is an affine function of  $y_1, y_2, \dots, y_K$  because  $\hat{x}$  is a linear function of  $1, y_1, y_2, \dots, y_K$ , and 1 can be regarded as a random variable.

Instead of trying to express that  $x - \hat{x}$  is orthogonal to every vector in  $S$ , it suffices to write enough linearly-independent equations to be able to solve for the  $\alpha_k$ s in (1.66). Since constant random variables are in  $S$  (by setting  $\alpha_1 = \alpha_2 = \dots = \alpha_K = 0$ ), we must have

$$\begin{aligned} 0 &\stackrel{(a)}{=} \langle x - \hat{x}, 1 \rangle \stackrel{(b)}{=} E[x - \hat{x}] \stackrel{(c)}{=} E[x] - E[\hat{x}] \\ &\stackrel{(d)}{=} E[x] - (\alpha_0 + \alpha_1 E[y_1] + \alpha_2 E[y_2] + \dots + \alpha_K E[y_K]), \end{aligned} \quad (1.67a)$$

where (a) follows from the desired orthogonality; (b) from (1.33); and (c) and (d) from linearity of the expectation. We also have that each  $y_k$  is in  $S$ , so by analogous steps

$$\begin{aligned} 0 &= \langle x - \hat{x}, y_k \rangle = E[(x - \hat{x})y_k^*] = E[xy_k^*] - E[\hat{x}y_k^*] \\ &= E[xy_k^*] - (\alpha_0 + \alpha_1 E[y_1 y_k^*] + \alpha_2 E[y_2 y_k^*] + \dots + \alpha_K E[y_K y_k^*]), \end{aligned} \quad (1.67b)$$

for  $k = 1, 2, \dots, K$ . Equations (1.67) can be rearranged using a matrix-vector product as

$$\begin{bmatrix} 1 & E[y_1] & E[y_2] & \dots & E[y_K] \\ E[y_1^*] & E[|y_1|^2] & E[y_2 y_1^*] & \dots & E[y_K y_1^*] \\ E[y_2^*] & E[y_1^* y_2] & E[|y_2|^2] & \dots & E[y_K y_2^*] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ E[y_K^*] & E[y_1 y_K^*] & E[y_2 y_K^*] & \dots & E[|y_K|^2] \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_K \end{bmatrix} = \begin{bmatrix} E[x] \\ E[xy_1^*] \\ E[xy_2^*] \\ \vdots \\ E[xy_K^*] \end{bmatrix}. \quad (1.68)$$

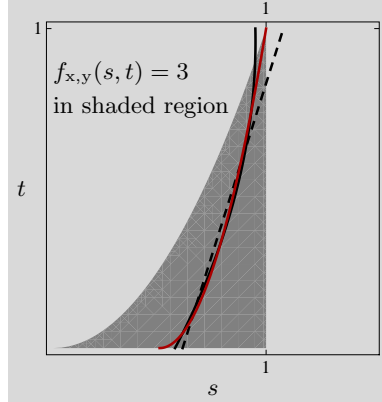
This system of equations will usually have a unique solution. The solution fails to be unique if and only if  $\{1, y_1, \dots, y_K\}$  is a linearly dependent set. In this case, (1.68) will have multiple solutions  $\{\alpha_k\}_{k=0}^K$ , but all the solutions yield the same estimator.

It is critical in this result that the set of estimators form a subspace, but this does not mean that the estimator must be an affine function of the observed data. For example, the estimator of  $x$  from a single scalar  $y$

$$\hat{x} = \beta_0 + \beta_1 y + \beta_2 y^2 + \dots + \beta_K y^K \quad (1.69)$$

fits the form of (1.66) with  $y_k$  set to  $y^k$ . Thus, assuming  $x$  has finite second moment and  $y$  has finite moments up to order  $2K$ , (1.68) can be used to optimize the estimator. Assuming  $y$  is real, (1.68) simplifies to

$$\begin{bmatrix} 1 & E[y] & E[y^2] & \dots & E[y^K] \\ E[y] & E[y^2] & E[y^3] & \dots & E[y^{K+1}] \\ E[y^2] & E[y^3] & E[y^4] & \dots & E[y^{K+2}] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ E[y^K] & E[y^{K+1}] & E[y^{K+2}] & \dots & E[y^{2K}] \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_K \end{bmatrix} = \begin{bmatrix} E[x] \\ E[xy^1] \\ E[xy^2] \\ \vdots \\ E[xy^K] \end{bmatrix}. \quad (1.70)$$



**Figure 1.17:** Minimum mean-squared error estimators of  $x$  from  $y$  from Examples 1.24 and 1.25. The joint distribution of  $x$  and  $y$  is uniform over the shaded region. The optimal estimates of the form  $\hat{x}_1 = \alpha_0 + \alpha_1 y$  (dashed line) and  $\hat{x}_2 = \beta_0 + \beta_1 y + \beta_2 y^2$  (solid curve) are derived in Example 1.24. The red curve is the optimal estimator  $\frac{1}{2}(1 + \sqrt{y})$  derived in Example 1.25.

EXAMPLE 1.24 (LINEAR MMSE ESTIMATORS) Suppose the joint distribution of  $x$  and  $y$  is uniform over the region shaded in Figure 1.17. Since the area of the shaded region is  $1/3$ , the joint PDF of  $x$  and  $y$  is

$$f_{x,y}(s, t) = \begin{cases} 3, & s \in [0, 1] \text{ and } t \in [0, s^2]; \\ 0, & \text{otherwise.} \end{cases}$$

We wish to find estimators of  $x$  from  $y$

$$\begin{aligned} \hat{x}_1 &= \alpha_0 + \alpha_1 y & \text{and} \\ \hat{x}_2 &= \beta_0 + \beta_1 y + \beta_2 y^2 \end{aligned}$$

that are optimal over the choices of coefficients  $\{\alpha_0, \alpha_1\}$  and  $\{\beta_0, \beta_1, \beta_2\}$ .

To form the system of equations (1.68), we make the following computations:

$$\begin{aligned} E[x] &= \int_0^1 \int_0^{s^2} 3s \, dt \, ds = \frac{3}{4}, \\ E[y] &= \int_0^1 \int_0^{s^2} 3t \, dt \, ds = \frac{3}{10}, \\ E[xy] &= \int_0^1 \int_0^{s^2} 3st \, dt \, ds = \frac{1}{4}, \\ E[y^2] &= \int_0^1 \int_0^{s^2} 3t^2 \, dt \, ds = \frac{1}{7}. \end{aligned}$$

Then  $\{\alpha_0, \alpha_1\}$  is determined by solving

$$\begin{bmatrix} 1 & 3/10 \\ 3/10 & 1/7 \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \end{bmatrix} = \begin{bmatrix} 3/4 \\ 1/4 \end{bmatrix}$$

to obtain  $\alpha_0 = 45/74$  and  $\alpha_1 = 35/74$ . The estimate  $\hat{x}_1$  is shown as a function of the observation  $y = t$  by the dashed line in Figure 1.17.

To find  $\{\beta_0, \beta_1, \beta_2\}$ , we require three additional moments:

$$E[xy^2] = \int_0^1 \int_0^{s^2} 3st^2 dt ds = \frac{1}{8},$$

$$E[y^3] = \int_0^1 \int_0^{s^2} 3t^3 dt ds = \frac{1}{12},$$

$$E[y^4] = \int_0^1 \int_0^{s^2} 3t^4 dt ds = \frac{3}{55}.$$

The system of equations (1.68) becomes

$$\begin{bmatrix} 1 & 3/10 & 1/7 \\ 3/10 & 1/7 & 1/12 \\ 1/7 & 1/12 & 3/55 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} 3/4 \\ 1/4 \\ 1/8 \end{bmatrix},$$

which yields  $\beta_0 = 12915/22558$ ,  $\beta_1 = 17745/22558$ , and  $\beta_2 = -4620/11279$ . The estimate  $\hat{x}_2$  is shown as a function of the observation  $y = t$  by the solid curve in Figure 1.17.

**General Optimal Estimation** The fact that estimators of the form (1.69) form a subspace hints at a more general fact: the set of *all* functions of a random variable form a subspace in a vector space of random variables. While this may seem surprising or counterintuitive, verification of the properties required by Definition 1.2 is trivial. The subspace of functions of a random variable is furthermore closed, so several properties of general (not-necessarily-linear) MMSE estimation follow from the projection theorem.

As the simplest example, the constant  $c$  that minimizes  $E[(x - c)^2]$  can be interpreted as the best estimator of  $x$  that depends on nothing random. We must have

$$0 \stackrel{(a)}{=} \langle x - c, 1 \rangle \stackrel{(b)}{=} E[x - c] \stackrel{(c)}{=} E[x] - c,$$

where (a) follows from the orthogonality of the error  $x - c$  to the deterministic function 1; (b) from (1.33); and (c) from linearity of the expectation. This derives the well-known fact that  $c = E[x]$  is the constant that minimizes  $E[(x - c)^2]$ ; see Appendix 1.C.3.

Now consider estimating  $x$  from observation  $y = t$ . Conditioning yields a valid probability law, so

$$\langle x, z \rangle = E[xz^* | y = t] \tag{1.71}$$

is an inner product on the set of random variables with finite conditional second moment given  $y = t$ . Orthogonality of error  $x - \hat{x}_{\text{MMSE}}(t)$  and any function of  $t$  under the inner product (1.71) yields

$$0 \stackrel{(a)}{=} E[x - \hat{x}_{\text{MMSE}}(t) | y = t] \stackrel{(b)}{=} E[x | y = t] - \hat{x}_{\text{MMSE}}(t),$$

where (a) follows from considering specifically the function 1; and (b) from  $\hat{x}_{\text{MMSE}}(t)$  being a (deterministic) function of  $t$ . Thus, the optimal estimate is the conditional mean:

$$\hat{x}_{\text{MMSE}}(t) = E[x | y = t], \tag{1.72a}$$

which is also written as

$$\hat{x}_{\text{MMSE}} = E[x|y]. \quad (1.72b)$$

EXAMPLE 1.25 (MMSE ESTIMATOR, EXAMPLE 1.24 CONT'D) Consider  $x$  and  $y$  jointly distributed as in Example 1.24 (see Figure 1.17). Given an observation  $y = t$ , the conditional distribution of  $x$  is uniform on  $[\sqrt{t}, 1]$ . The mean of this conditional distribution gives the optimal estimator

$$\hat{x}_{\text{MMSE}}(t) = \frac{1}{2} (1 + \sqrt{t}).$$

This optimal estimate is shown as a red curve in Figure 1.17.

### Orthogonality and Optimal Estimation of Random Vectors

Use of the inner product (1.33) has given us a geometric interpretation for *scalar* random variables with valuable ramifications for optimal estimation. One can define various inner products for random *vectors* as well. However, we will see that more useful estimation results come from generalizing the concept of orthogonality rather than from using a single inner product.

One valid inner product for complex random vectors of length  $N$  is obtained from the sum of inner products between components of the vectors,  $\langle x_n, y_n \rangle$ , using the inner product (1.33) between scalar random variables:

$$\langle x, y \rangle = \sum_{n=0}^{N-1} E[x_n y_n^*].$$

This is identical to the expectation of the standard inner product on  $\mathbb{C}^N$ , (1.20a), or  $\langle x, y \rangle = E[y^* x]$ .

With the projection theorem, one could optimize estimators of  $x$  from  $y^{(1)}, y^{(2)}, \dots, y^{(K)}$  of the form

$$\hat{x} = \alpha_0 \mathbf{1} + \alpha_1 y^{(1)} + \alpha_2 y^{(2)} + \dots + \alpha_K y^{(K)}, \quad (1.73)$$

where every component of  $\mathbf{1} \in \mathbb{C}^N$  is 1. This is exactly as in (1.66), but now each element of  $\{x, y^{(1)}, y^{(2)}, \dots, y^{(K)}\}$  is a vector rather than a scalar.<sup>14</sup> Optimal coefficients are determined by solving a system of equations analogous to (1.68).

A weakness of the estimator  $\hat{x}$  in (1.73) is that any single component of  $\hat{x}$  depends only on the corresponding components of  $\{y^{(1)}, y^{(2)}, \dots, y^{(K)}\}$ ; other dependencies are not exploited. To take a simple example, suppose  $x$  has the uniform distribution over the unit square  $[0, 1]^2$  and  $[y_1 \ y_2] = [x_2 \ x_1]$ . The vector  $x$  can be estimated perfectly from the vector  $y$ , but  $y_1$  is useless in estimating  $x_1$  and  $y_2$  is useless in estimating  $x_2$ . Thus estimators more general than (1.73) are commonly used.

<sup>14</sup>Superscripts are introduced so we do not confuse indexing of the set with indexing of the components of a single vector.

**Linear Estimation** Let  $\mathbf{x}$  be a  $\mathbb{C}^N$ -valued random vector, and let  $\mathbf{y}$  be a  $\mathbb{C}^M$ -valued random vector. Suppose all components of both vectors have finite second moments. Consider estimator of  $\mathbf{x}$  from  $\mathbf{y}$  given by<sup>15</sup>

$$\hat{\mathbf{x}} = A\mathbf{y}, \quad (1.74)$$

where  $A \in \mathbb{C}^{N \times M}$  is a constant matrix to be designed to minimize the MSE  $E[\|\mathbf{x} - \hat{\mathbf{x}}\|^2]$ . Note that unlike in (1.73), every component of  $\hat{\mathbf{x}}$  depends on every component of  $\mathbf{y}$ .

Since each row of  $A$  determines a different component of  $\hat{\mathbf{x}}$  and the MSE decouples across components as

$$E[\|\mathbf{x} - \hat{\mathbf{x}}\|^2] = \sum_{n=0}^{N-1} E[|x_n - \hat{x}_n|^2],$$

we can consider the design of each row of  $A$  separately. Then for any fixed  $n \in \{0, 1, \dots, N-1\}$ , the minimization of  $E[|x_n - \hat{x}_n|^2]$  through the choice of the  $n$ th row of  $A$  is a problem we have already solved: it is the scalar linear MMSE estimation problem. The solution is characterized by orthogonality of the error and the data as in (1.67).

The orthogonality of  $n$ th error component  $x_n - \hat{x}_n$  and  $m$ th data component  $y_m$  can be expressed as

$$0 \stackrel{(a)}{=} E[(x_n - \hat{x}_n)y_m^*] \stackrel{(b)}{=} E[(x_n - a_n^T \mathbf{y})y_m^*],$$

where (a) follows from the inner product (1.33); and (b) introduces  $a_n^T$  as the  $n$ th row of  $A$ . Gathering these equations for  $m = 0, 1, \dots, M-1$  into one row gives

$$\mathbf{0}_{1 \times M} = E[(x_n - a_n^T \mathbf{y})\mathbf{y}^*], \quad n = 0, 1, \dots, N-1.$$

Now stacking these equations into a matrix gives

$$\mathbf{0}_{N \times M} = E[(\mathbf{x} - A\mathbf{y})\mathbf{y}^*]. \quad (1.75)$$

Using linearity of expectation, a necessary and sufficient condition for optimality is thus

$$E[\mathbf{x}\mathbf{y}^*] = AE[\mathbf{y}\mathbf{y}^*]. \quad (1.76)$$

In most cases,  $E[\mathbf{y}\mathbf{y}^*]$  is invertible; the optimal estimator is then

$$\hat{\mathbf{x}}_{\text{MMSE}} = E[\mathbf{x}\mathbf{y}^*] (E[\mathbf{y}\mathbf{y}^*])^{-1} \mathbf{y}. \quad (1.77)$$

When  $E[\mathbf{y}\mathbf{y}^*]$  is not invertible, solutions to (1.76) are not unique but yield the same estimator.

**Orthogonality of Random Vectors** Inspired by the usefulness of (1.75) in optimal estimation, we define a new orthogonality concept for random vectors.

<sup>15</sup>In contrast to estimators (1.66) and (1.73), we have omitted a constant term from the estimator. There is no loss of generality because one can augment  $\mathbf{y}$  with a constant random variable.



**DEFINITION 1.32 (ORTHOGONAL RANDOM VECTORS)** Random vectors  $\mathbf{x}$  and  $\mathbf{y}$  are said to be orthogonal when  $E[\mathbf{x}\mathbf{y}^*] = \mathbf{0}$ .

Note that  $E[\mathbf{x}\mathbf{y}^*]$  is not an inner product because it is not a scalar (except in the degenerate case where the random vectors have dimension 1 and are thus scalar random variables). Instead, random vectors  $\mathbf{x}$  and  $\mathbf{y}$  are orthogonal when every combination of components are orthogonal under inner product (1.33):

$$E[\mathbf{x}_n \mathbf{y}_m^*] = 0 \quad \text{for every } m \text{ and } n.$$

In (1.75) we have an instance of a more general fact: Any time an estimator  $\hat{\mathbf{x}}$  of random vector  $\mathbf{x}$  is optimized over a closed subspace of possible estimators  $S$ , the optimal estimator will be determined by  $\mathbf{x} - \hat{\mathbf{x}} \perp S$  under the sense of orthogonality in Definition 1.32. We will apply this to optimal LMMSE estimation of discrete-time random processes in Chapter 2.

## 1.5 Bases and Frames

The variety of bases and frames for sequences in  $\ell^2(\mathbb{Z})$  and functions in  $\mathcal{L}^2(\mathbb{R})$  is at the heart of this book. In this section, we develop general properties of bases and frames, with an emphasis on representing vectors in Hilbert spaces using bases. Bases will come in two flavors, orthonormal and biorthogonal; analogously, frames will come in two flavors, tight and general. In later chapters, we will see that the choice of a basis or frame can have dramatic effects on the computational complexity, stability, and approximation accuracy of signal expansions.

Prominent in the developments of Section 1.4 were closed subspaces: We saw best approximation in a closed subspace, projection onto a closed subspace, and direct sum decomposition into a pair of closed subspaces. In this section, we will see that a basis induces a direct sum decomposition into a possibly-infinite number of one-dimensional subspaces; and bases, especially orthonormal ones, facilitate the computations of projections and approximations. These developments reduce our level of abstraction and bring us closer to computational tools for signal processing. Specifically, Section 1.5.5 shows how representations with bases replace general vector space computations with matrix computations, albeit possibly infinite ones.

### 1.5.1 Bases and Riesz Bases

A basis is a set of vectors that is used to uniquely represent any vector in a vector space as a linear combination of basis elements. It is of minimal size in that it is a linearly independent set. The definition applies in any normed vector space, including any Hilbert space (where the norm is induced by an inner product).

**DEFINITION 1.33 (BASIS)** The set of vectors  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}} \subset V$ , where  $\mathcal{K}$  is finite or countably infinite, is called a basis for the normed vector space  $V$  when

- (i) it is *linearly independent*; and
- (ii) it is *complete* in  $V$ , meaning

$$V = \overline{\text{span}}(\Phi). \quad (1.78)$$

In this definition, the closure of the span is needed to allow linear combinations with infinitely-many terms.<sup>16</sup> The completeness requirement implies that any  $x \in H$  has an *expansion* with respect to the basis  $\Phi$  of the form

$$x = \sum_{k \in \mathcal{K}} \alpha_k \varphi_k, \quad (1.79)$$

while the linear independence requirement implies that the expansion is unique: Given another expansion  $x = \sum_{k \in \mathcal{K}} \beta_k \varphi_k$ , subtracting it from (1.79) gives  $0 = \sum_{k \in \mathcal{K}} (\alpha_k - \beta_k) \varphi_k$ , so linear independence implies  $\alpha_k - \beta_k = 0$  for all  $k \in \mathcal{K}$ . The coefficients  $\{\alpha_k\}_{k \in \mathcal{K}}$  are called the *expansion coefficients*<sup>17</sup> of  $x$  with respect to the basis  $\Phi$ .

From Definition 1.6, it is clear that when a vector space has finite dimension  $N$ , its bases contain precisely  $N$  elements. The infinite-dimensional Hilbert spaces that we consider have countably-infinite bases because they are separable (see Section 1.3.2).

EXAMPLE 1.26 (STANDARD BASIS FOR  $\mathbb{C}^N$ ) The standard basis for  $\mathbb{R}^2$  was introduced in Section 1.1. It is easily extended to  $\mathbb{C}^N$  (or  $\mathbb{R}^N$ ) with

$$e_k = \left[ \underbrace{0 \ 0 \ \dots \ 0}_{k \text{ 0s}} \ 1 \ \underbrace{0 \ 0 \ \dots \ 0}_{(N-k-1) \text{ 0s}} \right]^T, \quad k = 0, 1, \dots, N-1.$$

The set  $\{e_k\}_{k=0}^{N-1}$  is both linearly independent and complete, and it is thus a basis. For completeness, note that any vector  $v = [v_0 \ v_1 \ \dots \ v_{N-1}]^T \in \mathbb{C}^N$  is precisely the finite linear combination  $v = \sum_{k=0}^{N-1} v_k e_k$ ; the closure in (1.78) is not needed.

EXAMPLE 1.27 (STANDARD BASIS FOR  $\mathbb{C}^{\mathbb{Z}}$ ) The standard basis concept extends also to some normed vector spaces of complex-valued sequences over  $\mathbb{Z}$ . Consider  $E = \{e_k\}_{k \in \mathbb{Z}}$  where  $e_k \in \mathbb{C}^{\mathbb{Z}}$  is the sequence that is 0 except for a 1 in the  $k$ -indexed position. The set  $E$  is clearly linearly independent, but whether it is complete depends on the norm.

The set  $E$  is complete under the  $\ell^p$  norm with  $p \in [1, \infty)$ , meaning that  $\ell^p(\mathbb{Z}) = \overline{\text{span}}(E)$  when the closure of the span is defined with the  $\ell^p$  norm.

<sup>16</sup>Recall that the span of an infinite set of vectors is defined as the set of all finite linear combinations (Definition 1.4).

<sup>17</sup>Expansion coefficients are sometimes called *Fourier coefficients* or *generalized Fourier coefficients*, but we will avoid these terms except when the expansions are with respect to the specific bases that yield the various Fourier transforms. They are also called *transform coefficients* or *subband coefficients* in the source coding literature.

To show this, we establish  $\overline{\text{span}}(E) \subseteq \ell^p(\mathbb{Z})$  and  $\ell^p(\mathbb{Z}) \subseteq \overline{\text{span}}(E)$ . The first inclusion,  $\overline{\text{span}}(E) \subseteq \ell^p(\mathbb{Z})$ , holds because  $\ell^p(\mathbb{Z})$  is a complete vector space; see Section 1.3.2. It remains to show  $\ell^p(\mathbb{Z}) \subseteq \overline{\text{span}}(E)$ . The meaning of  $x \in \ell^p(\mathbb{Z})$  is that  $\sum_{n \in \mathbb{Z}} |x_n|^p$  is convergent. Thus,

$$\lim_{M \rightarrow \infty} \left( \sum_{n=-\infty}^{-M-1} |x_n|^p + \sum_{n=M+1}^{\infty} |x_n|^p \right) = 0.$$

This limit shows that the sequence  $y_M = \sum_{n=-M}^M x_n e_n$ ,  $M = 0, 1, \dots$ , converges to  $x$  under the  $\ell^p(\mathbb{Z})$  norm. Since every  $y_M$  is in  $\text{span}(E)$ , the limit of the sequence  $x$  is in  $\overline{\text{span}}(E)$ . Thus  $\ell^p(\mathbb{Z}) \subseteq \overline{\text{span}}(E)$ .

Changing the norm changes the meaning of the closure of the span and thus can make  $E$  not be a basis. The set  $E$  is not complete under the  $\ell^\infty$  norm since  $\ell^\infty(\mathbb{Z}) \not\subseteq \overline{\text{span}}(E)$ . To show this, let  $x$  be the all 1s sequence, which is in  $\ell^\infty(\mathbb{Z})$ . Every sequence in  $\text{span}(E)$  has a finite number of nonzero entries and is therefore at least distance 1 from  $x$  under the  $\ell^\infty$  norm. Therefore  $x$  is not in  $\overline{\text{span}}(E)$ .

**Riesz Bases** While the previous example provides an important note of caution on the meaning of a basis, dependence on the choice of norm is not our focus. In fact, we will primarily focus on the  $\ell^2$  and  $\mathcal{L}^2$  Hilbert space norms. The more important complication with infinite-dimensional spaces is that a basis can be prohibitively ill-suited to numerical computations. Specifically, it is not practical to allow coefficients in a linear combination to be unbounded or to require very small coefficients to be distinguished from zero; the concept of a Riesz basis restricts bases to avoid these pitfalls.

**DEFINITION 1.34 (RIESZ BASIS)** The set of vectors  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}} \subset H$ , where  $\mathcal{K}$  is finite or countably infinite, is called a Riesz basis for Hilbert space  $H$  when

- (i) it is a *basis* for  $H$ ; and
- (ii) there exist strictly positive constants  $\lambda_{\min}$  and  $\lambda_{\max}$  such that, for any  $x$  in  $H$ , the expansion of  $x$  with respect to the basis  $\Phi$ ,  $x = \sum_{k \in \mathcal{K}} \alpha_k \varphi_k$ , satisfies

$$\lambda_{\min} \|x\|^2 \leq \sum_{k \in \mathcal{K}} |\alpha_k|^2 \leq \lambda_{\max} \|x\|^2. \quad (1.80)$$

In  $\mathbb{C}^N$  or  $\ell^2(\mathbb{Z})$ , the standard basis is a Riesz basis with  $\lambda_{\min} = \lambda_{\max} = 1$  (see Exercise 1.35). Conversely, Riesz bases with  $\lambda_{\min} = \lambda_{\max} = 1$  are orthonormal bases, as developed in Section 1.5.2. As we introduce a variety of bases for different purposes, it will be a virtue to have  $\lambda_{\min} \approx \lambda_{\max}$ , though we may relax this requirement to achieve other objectives.

EXAMPLE 1.28 (RIESZ BASES IN  $\mathbb{R}^2$ ) Any two vectors  $\varphi_0$  and  $\varphi_1$  are a basis for  $\mathbb{R}^2$  as long as there is no scalar  $\alpha$  such that  $\varphi_1 = \alpha\varphi_0$ . We fix  $\varphi_0 = e_0$  and vary  $\varphi_1$  in two ways to illustrate deviations from the standard basis:

- (i) Let  $\varphi_1 = ae_1$  with  $a \in (0, \infty)$ , as illustrated in Figure 1.18(a). The unique expansion of  $\begin{bmatrix} x_0 & x_1 \end{bmatrix}^T$  is then

$$x = x_0\varphi_0 + (x_1/a)\varphi_1 = \alpha_0\varphi_0 + \alpha_1\varphi_1.$$

The largest  $\lambda_{\min}$  such that (1.80) holds is

$$\lambda_{\min} = \inf_{x \in \mathbb{R}^2} \frac{x_0^2 + (x_1/a)^2}{x_0^2 + x_1^2} = \begin{cases} 1, & \text{for } a \in (0, 1]; \\ 1/a^2, & \text{for } a \in (1, \infty). \end{cases}$$

This means that by making  $a$  very large, the basis becomes numerically ill-conditioned in the sense that there are nonzero vectors  $x$  with very small expansion coefficients.

Similarly, the smallest  $\lambda_{\max}$  such that (1.80) holds is

$$\lambda_{\max} = \sup_{x \in \mathbb{R}^2} \frac{x_0^2 + (x_1/a)^2}{x_0^2 + x_1^2} = \begin{cases} 1/a^2, & \text{for } a \in (0, 1]; \\ 1, & \text{for } a \in (1, \infty). \end{cases}$$

This means that by making  $a$  close to zero, the basis becomes numerically ill-conditioned in the sense that there are vectors  $x$  with very large expansion coefficients. Figure 1.18(c) shows  $\lambda_{\min}$ ,  $\lambda_{\max}$  as functions of  $a$ .

- (ii) Let  $\varphi_1 = [\cos \theta \quad \sin \theta]^T$  with  $\theta \in (0, \pi/2]$ , as illustrated in Figure 1.18(b). The unique expansion of  $\begin{bmatrix} x_0 & x_1 \end{bmatrix}^T$  is then

$$x = (x_0 - \cot \theta x_1)\varphi_0 + (\csc \theta x_1)\varphi_1 = \alpha_0\varphi_0 + \alpha_1\varphi_1.$$

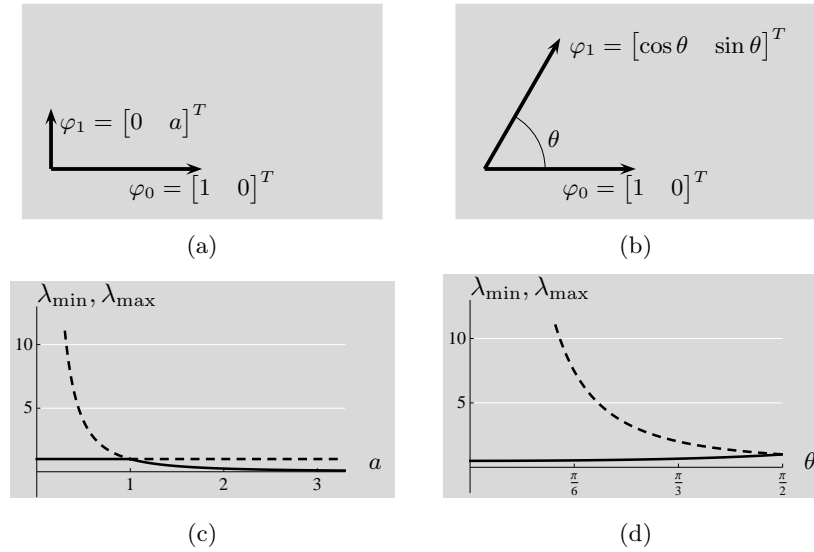
Using trigonometric identities, one can show that the largest  $\lambda_{\min}$  and smallest  $\lambda_{\max}$  such that (1.80) holds are

$$\lambda_{\min} = \frac{1}{2} \sec^2(\theta/2) \quad \text{and} \quad \lambda_{\max} = \frac{1}{2} \csc^2(\theta/2),$$

shown in Figure 1.18(d) as functions of  $\theta$ . The numerical conditioning is ideal when  $\theta = \pi/2$ , in which case  $\{\varphi_0, \varphi_1\}$  is the standard basis, while it is extremely poor for small  $\theta$ , resulting in very large expansion coefficients.

The previous example illustrates two ways of deviating from the standard basis: lacking unit norm and lacking orthogonality. While the effects of these deviations can make numerical conditioning arbitrarily bad, any basis of a finite-dimensional Hilbert space is a Riesz basis. (In the first part of the example,  $a$  must be nonzero for  $\{\varphi_0, \varphi_1\}$  to be a basis, and this keeps  $\lambda_{\min}$  strictly positive and  $\lambda_{\max}$  finite. Similarly, in the second part  $\theta$  must be nonzero, and this keeps  $\lambda_{\max}$  finite.) The following infinite-dimensional examples show that some bases are not Riesz bases.

EXAMPLE 1.29 (BASES THAT ARE NOT RIESZ BASES)



**Figure 1.18:** Two families of bases in  $\mathbb{R}^2$  that deviate from the standard basis  $\{e_0, e_1\}$  and their Riesz basis stability constants  $\lambda_{\min}$  (solid) and  $\lambda_{\max}$  (dashed). (a)  $\varphi_1$  is orthogonal to  $\varphi_0$  but not necessarily of unit length. (b)  $\varphi_1$  is of unit length but not necessarily orthogonal to  $\varphi_0$ . (c)  $\lambda_{\min}$  and  $\lambda_{\max}$  for the basis in (a) as a function of  $a$ . (d)  $\lambda_{\min}$  and  $\lambda_{\max}$  for the basis in (b) as a function of  $\theta$ .

- (i) Consider the following scaled version of the standard basis in  $\ell^2(\mathbb{Z})$ :

$$\varphi_k = (1 + |k|)^{-1} e_k, \quad k \in \mathbb{Z}.$$

The ratio of lengths of elements  $\|\varphi_k\|/\|\varphi_0\|$  is unbounded, so this is intuitively similar to letting  $a \rightarrow 0$  or  $a \rightarrow \infty$  in Example 1.28(i). The set  $\Phi = \{\varphi_k\}_{k \in \mathbb{Z}}$  is a basis for  $\ell^2(\mathbb{Z})$ , but it is not a Riesz basis.

To prove that  $\Phi$  is not a Riesz basis, we show that no finite  $\lambda_{\max}$  satisfies (1.80). Suppose there is a finite  $\lambda_{\max}$  such that (1.80) holds for all  $x \in \ell^2(\mathbb{Z})$ . Then we can choose an integer  $M > \sqrt{\lambda_{\max}}$  and let  $x \in \ell^2(\mathbb{Z})$  be the sequence that is 0 except for a 1 in the  $M$ -indexed position. The unique representation of this  $x$  using the basis  $\Phi$  is  $x = (1 + |M|)\varphi_M$ ; that is, the coefficients in the expansion are

$$\alpha_k = \begin{cases} 1 + |M|, & \text{for } k = M; \\ 0, & \text{otherwise.} \end{cases}$$

The second inequality of (1.80) is contradicted, so there does not exist the desired finite  $\lambda_{\max}$ .

- (ii) Consider the following vectors defined using the standard basis in  $\ell^2(\mathbb{N})$ :

$$\varphi_k = \sum_{i=0}^k (i+1)^{-1/2} e_i, \quad k \in \mathbb{N}.$$

The angle between consecutive elements approaches zero as  $k \rightarrow \infty$  because  $\langle \varphi_{k+1}, \varphi_k \rangle \rightarrow \|\varphi_{k+1}\| \|\varphi_k\| = 1$ , so this is intuitively similar to letting  $\theta \rightarrow 0$  in Example 1.28(ii). Proving that the set  $\Phi = \{\varphi_k\}_{k \in \mathbb{N}}$  is a basis for  $\ell^2(\mathbb{N})$  but is not a Riesz basis is left for Exercise 1.36.

In the subsequent developments, it will be desirable for all bases to be Riesz bases.

**Operators Associated with Bases** Given a Riesz basis  $\{\varphi_k\}_{k \in \mathcal{K}}$ , the expansion formula (1.79) can be viewed as mapping a coefficient sequence  $\alpha$  to a vector  $x$ . This mapping is clearly linear. Let us suppose that the coefficient sequence has finite  $\ell^2(\mathcal{K})$  norm. The first inequality of (1.80) implies that the vector  $x$  given by (1.79) has finite norm, at most  $\|\alpha\|/\sqrt{\lambda_{\min}}$ , and is thus legitimately in  $H$ .

**DEFINITION 1.35 (BASIS SYNTHESIS OPERATOR)** Given a Riesz basis  $\{\varphi_k\}_{k \in \mathcal{K}}$ , the synthesis operator associated with it is

$$\Phi : \ell^2(\mathcal{K}) \rightarrow H, \quad \text{with} \quad \Phi \alpha = \sum_{k \in \mathcal{K}} \alpha_k \varphi_k. \quad (1.81)$$

The second inequality of (1.80) implies that the norm of this linear operator is at most  $\sqrt{\lambda_{\max}}$ ; the operator  $\Phi$  is thus not only linear but bounded as well.

The adjoint of  $\Phi$  maps from  $H$  to a sequence in  $\ell^2(\mathcal{K})$ . To derive the adjoint, consider the following computation for arbitrary  $\alpha \in \ell^2(\mathcal{K})$  and  $y \in H$ :

$$\langle \Phi \alpha, y \rangle \stackrel{(a)}{=} \left\langle \sum_{k \in \mathcal{K}} \alpha_k \varphi_k, y \right\rangle \stackrel{(b)}{=} \sum_{k \in \mathcal{K}} \alpha_k \langle \varphi_k, y \rangle \stackrel{(c)}{=} \sum_{k \in \mathcal{K}} \alpha_k \langle y, \varphi_k \rangle^*,$$

where (a) follows from (1.81); (b) from the linearity in the first argument of the inner product; and (c) from the Hermitian symmetry of the inner product. The final expression is the  $\ell^2(\mathcal{K})$  inner product between  $\alpha$  and a sequence of inner products  $\{\langle y, \varphi_k \rangle\}_{k \in \mathcal{K}}$ . The adjoint is called the analysis operator:

**DEFINITION 1.36 (BASIS ANALYSIS OPERATOR)** Given a Riesz basis  $\{\varphi_k\}_{k \in \mathcal{K}}$ , the analysis operator associated with it is

$$\Phi^* : H \rightarrow \ell^2(\mathcal{K}), \quad \text{with} \quad (\Phi^* x)_k = \langle x, \varphi_k \rangle, \quad k \in \mathcal{K}. \quad (1.82)$$

Equation 1.82 holds since  $\langle \Phi \alpha, y \rangle_H = \langle \alpha, \Phi^* y \rangle_{\ell^2}$ . The norm of the analysis operator is also at most  $\sqrt{\lambda_{\max}}$  because  $\|A\| = \|A^*\|$  for all bounded linear operators  $A$ .

## 1.5.2 Orthonormal Bases

An *orthonormal basis* is a basis of orthogonal, unit-norm vectors:

**DEFINITION 1.37 (ORTHONORMAL BASIS)** The set of vectors  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}} \subset H$ , where  $\mathcal{K}$  is finite or countably infinite, is called an orthonormal basis for the Hilbert space  $H$  when

- (i) it is a *basis* for  $H$ ; and
- (ii) it is *orthonormal*,

$$\langle \varphi_i, \varphi_k \rangle = \delta_{i-k} \quad \text{for every } i, k \in \mathcal{K}. \quad (1.83)$$

Since orthonormality implies linear independence, we could alternatively say that a set  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}} \subset H$  satisfying (1.83) is an orthonormal basis whenever it is complete, that is,  $\overline{\text{span}}(\Phi) = H$ .

Standard bases are orthonormal bases. Two more examples follow, and we will see many more examples throughout the book.

**EXAMPLE 1.30 (FINITE-DIMENSIONAL ORTHONORMAL BASIS)** The vectors

$$\varphi_0 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \quad \varphi_1 = \frac{1}{\sqrt{6}} \begin{bmatrix} -1 \\ 1 \\ 2 \end{bmatrix}, \quad \text{and} \quad \varphi_2 = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$$

are orthonormal, as can be verified by direct computation. Since three linearly independent vectors in  $\mathbb{C}^3$  always form a basis for  $\mathbb{C}^3$ ,  $\{\varphi_0, \varphi_1, \varphi_2\}$  is an orthonormal basis for  $\mathbb{C}^3$ .

**EXAMPLE 1.31 (ORTHONORMAL BASIS OF COSINE FUNCTIONS)** Consider  $\Phi = \{\varphi_k\}_{k \in \mathbb{N}} \subset \mathcal{L}^2([-\frac{1}{2}, \frac{1}{2}])$  defined in (1.21). The first three functions in this set were shown in Figure 1.5. Example 1.6(iii) showed that  $\Phi$  satisfies the orthonormality condition (1.83). Since orthonormality also implies linear independence,  $\Phi$  is an orthonormal basis for  $S = \overline{\text{span}}(\Phi)$ . (Remember that  $S$  is itself a Hilbert space.) The set  $\Phi$  is not, however, an orthonormal basis for  $\mathcal{L}^2([-\frac{1}{2}, \frac{1}{2}])$  because  $S$  is a proper subspace of  $\mathcal{L}^2([-\frac{1}{2}, \frac{1}{2}])$ ; for example, no odd functions are in  $S$ .

**Expansion and Inner Product Computation** Expansion coefficients with respect to an orthonormal basis are obtained by using the same basis for signal analysis.

**THEOREM 1.38 (ORTHONORMAL BASIS EXPANSIONS)** Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}}$  be an orthonormal basis for Hilbert space  $H$ . The unique expansion with respect to  $\Phi$  of any  $x$  in  $H$  has expansion coefficients

$$\alpha_k = \langle x, \varphi_k \rangle \quad \text{for } k \in \mathcal{K}, \quad \text{or}, \quad (1.84a)$$

$$\alpha = \Phi^* x. \quad (1.84b)$$

Synthesis with these coefficients yields

$$x = \sum_{k \in \mathcal{K}} \langle x, \varphi_k \rangle \varphi_k \quad (1.85a)$$

$$= \Phi \alpha = \Phi \Phi^* x. \quad (1.85b)$$

*Proof.* The existence of a unique linear combination of the form (1.79) is guaranteed by  $\Phi$  being a basis. The validity of (1.85a) with coefficients (1.84a) follows from the following computation:

$$\langle x, \varphi_k \rangle \stackrel{(a)}{=} \langle \sum_{i \in \mathcal{K}} \alpha_i \varphi_i, \varphi_k \rangle \stackrel{(b)}{=} \sum_{i \in \mathcal{K}} \alpha_i \langle \varphi_i, \varphi_k \rangle \stackrel{(c)}{=} \sum_{i \in \mathcal{K}} \alpha_i \delta_{i-k} \stackrel{(d)}{=} \alpha_k,$$

where (a) follows from (1.79); (b) from the linearity in the first argument of the inner product; (c) from the orthonormality of the set  $\Phi$ , (1.83); and (d) from the definition of the Kronecker delta sequence, (1.9).

The expressions (1.84b) and (1.85b) are equivalent to (1.84a) and (1.85a) using the operators defined in (1.81) and (1.82).

Since (1.85b) holds for all  $x$  in  $H$ ,

$$\Phi \Phi^* = I \quad \text{on } H. \quad (1.86)$$

This leads to the frequently-used properties<sup>18</sup> in the following theorem:

**THEOREM 1.39 (PARSEVAL'S EQUALITIES)** Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}}$  be an orthonormal basis for Hilbert space  $H$ . Expansion with coefficients (1.84) satisfies

$$\|x\|^2 = \sum_{k \in \mathcal{K}} |\langle x, \varphi_k \rangle|^2 \quad (1.87a)$$

$$= \|\Phi^* x\|^2 = \|\alpha\|^2. \quad (1.87b)$$

More generally,

$$\langle x, y \rangle = \sum_{k \in \mathcal{K}} \langle x, \varphi_k \rangle \langle y, \varphi_k \rangle^* \quad (1.88a)$$

$$= \langle \Phi^* x, \Phi^* y \rangle = \langle \alpha, \beta \rangle. \quad (1.88b)$$

*Proof.* Recall the equivalence of (1.50) and (1.52a). Thus (1.86) is equivalent to (1.88b). Setting  $x = y$  in (1.88b) yields (1.87b). Equalities (1.87a) and (1.88a) are the same facts expanded with the definition of  $\Phi^*$ .

Proving this using operator notation and properties in (1.86) is much less tedious than the direct proof. To see this on the example of (1.88), write:

$$\langle x, y \rangle \stackrel{(a)}{=} \langle \sum_{k \in \mathcal{K}} \langle x, \varphi_k \rangle \varphi_k, y \rangle \stackrel{(b)}{=} \sum_{k \in \mathcal{K}} \langle x, \varphi_k \rangle \langle \varphi_k, y \rangle \stackrel{(c)}{=} \sum_{k \in \mathcal{K}} \langle x, \varphi_k \rangle \langle y, \varphi_k \rangle^*,$$

<sup>18</sup>The first of these, (1.87a), is the one most often referred to as the Parseval's equality.



## 1.5. Bases and Frames

73

where (a) follows from expanding  $x$  with (1.85a); (b) from the linearity in the first argument of the inner product; and (c) from the Hermitian symmetry of the inner product.

The simple equality (1.88) captures an important role played by any orthonormal basis: it turns an abstract inner product computation into a computation with sequences. When  $x = \sum_{k \in \mathcal{K}} \alpha_k \varphi_k$  and  $y = \sum_{k \in \mathcal{K}} \beta_k \varphi_k$  as in the theorem,

$$\langle x, y \rangle_H = \langle \alpha, \beta \rangle_{\ell^2(\mathcal{K})} = \sum_{k \in \mathcal{K}} \alpha_k \beta_k^*, \quad (1.89)$$

where the final computation is an  $\ell^2(\mathcal{K})$  inner product even though the first inner product is in an arbitrary Hilbert space. We may view this more concretely with matrix multiplication as

$$\langle x, y \rangle = \beta^* \alpha, \quad (1.90)$$

where  $\alpha$  and  $\beta$  are column vectors.

**EXAMPLE 1.32 (INNER PRODUCT COMPUTATION BY EXPANSION SEQUENCES)** Let  $\alpha$  and  $\beta$  be sequences in  $\ell^2(\mathbb{N})$ . Then the functions

$$x(t) = \alpha_0 + \sum_{k=1}^{\infty} \alpha_k \sqrt{2} \cos(2\pi kt), \quad (1.91a)$$

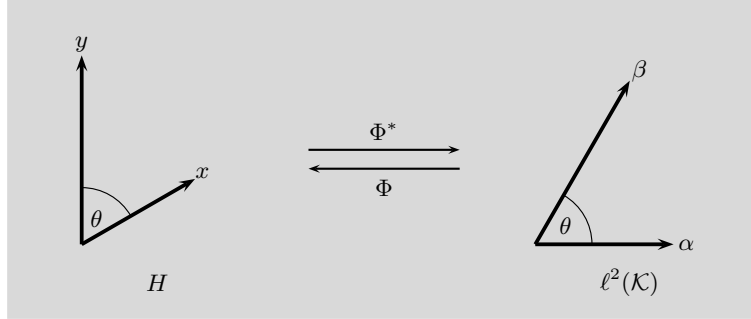
$$y(t) = \beta_0 + \sum_{k=1}^{\infty} \beta_k \sqrt{2} \cos(2\pi kt), \quad (1.91b)$$

are in  $\mathcal{L}^2([-\frac{1}{2}, \frac{1}{2}])$ , and their inner product can be simplified as follows:

$$\begin{aligned} \langle x, y \rangle &= \int_{-1/2}^{1/2} \left( \alpha_0 + \sum_{k=1}^{\infty} \alpha_k \sqrt{2} \cos(2\pi kt) \right) \left( \beta_0^* + \sum_{\ell=1}^{\infty} \beta_{\ell}^* \sqrt{2} \cos(2\pi \ell t) \right) dt \\ &= \alpha_0 \beta_0^* + \underbrace{\alpha_0 \sum_{\ell=1}^{\infty} \beta_{\ell}^* \int_{-1/2}^{1/2} \sqrt{2} \cos(2\pi \ell t) dt}_{=0} + \underbrace{\beta_0^* \sum_{k=1}^{\infty} \alpha_k \int_{-1/2}^{1/2} \sqrt{2} \cos(2\pi kt) dt}_{=0} \\ &\quad + \sum_{k=1}^{\infty} \sum_{\ell=1}^{\infty} \alpha_k \beta_{\ell}^* \underbrace{\int_{-1/2}^{1/2} 2 \cos(2\pi kt) \cos(2\pi \ell t) dt}_{=\delta_{k-\ell}} \\ &= \alpha_0 \beta_0^* + \sum_{k=1}^{\infty} \sum_{\ell=1}^{\infty} \alpha_k \beta_{\ell}^* \delta_{k-\ell} \stackrel{(a)}{=} \alpha_0 \beta_0^* + \sum_{k=1}^{\infty} \alpha_k \beta_k^* = \sum_{k=0}^{\infty} \alpha_k \beta_k^* \\ &\stackrel{(b)}{=} \langle \alpha, \beta \rangle, \end{aligned}$$

where (a) follows from the definition of the Kronecker delta sequence, (1.9); and (b) from the definition of the  $\ell^2$  inner product between sequences.

Recalling the orthonormal basis from Example 1.31, what we have shown is that a more complicated (integral) inner product in  $\mathcal{L}^2(\mathbb{R})$  can be computed via a simpler (series) inner product between expansion coefficients in  $\ell^2(\mathbb{Z})$ .



**Figure 1.19:** Conceptual illustration of the isometry between a separable Hilbert space  $H$  and sequence space  $\ell^2(\mathcal{K})$  induced by an orthonormal basis  $\Phi$ . It preserves geometry as  $\langle x, y \rangle = \langle \alpha, \beta \rangle$ .

**Unitary Synthesis and Analysis** We will show

$$\Phi^* \Phi = I \quad \text{on } \ell^2(\mathcal{K}). \quad (1.92)$$

Combined with (1.86), this establishes that the analysis and synthesis operators associated with an orthonormal basis are unitary.

To verify (1.92), make the following computation for any sequence  $\alpha$  in  $\ell^2(\mathcal{K})$ :

$$\begin{aligned} \Phi^* \Phi \alpha &\stackrel{(a)}{=} \Phi^* \sum_{i \in \mathcal{K}} \alpha_i \varphi_i \stackrel{(b)}{=} \left\{ \langle \sum_{i \in \mathcal{K}} \alpha_i \varphi_i, \varphi_k \rangle \right\}_{k \in \mathcal{K}} \stackrel{(c)}{=} \left\{ \sum_{i \in \mathcal{K}} \alpha_i \langle \varphi_i, \varphi_k \rangle \right\}_{k \in \mathcal{K}} \\ &\stackrel{(d)}{=} \left\{ \sum_{i \in \mathcal{K}} \alpha_i \delta_{i-k} \right\}_{k \in \mathcal{K}} \stackrel{(e)}{=} \{\alpha_k\}_{k \in \mathcal{K}} = \alpha, \end{aligned} \quad (1.93)$$

where (a) follows from (1.81); (b) from (1.82); (c) from the linearity in the first argument of the inner product; (d) from orthonormality the set  $\{\varphi_k\}_{k \in \mathcal{K}}$ , (1.83); and (e) from the definition of the Kronecker delta sequence, (1.9).

**Isometry of Separable Hilbert Spaces and  $\ell^2(\mathcal{K})$**  The fact that the synthesis and analysis operators  $\Phi$  and  $\Phi^*$  associated with an orthonormal basis are unitary leads to key intuitions about separable Hilbert spaces. A unitary operator between Hilbert spaces puts Hilbert spaces in one-to-one correspondence while preserving the geometries (that is, inner products) in the spaces. Since Hilbert spaces that we consider are separable, they contain orthonormal bases. Therefore, these Hilbert spaces can all be put in one-to-one correspondence with  $\mathbb{C}^N$  if they are finite-dimensional or with  $\ell^2(\mathbb{Z})$  if they are infinite-dimensional, as illustrated in Figure 1.19. (The notation  $\ell^2(\mathcal{K})$ , with  $\mathcal{K}$  finite or countable infinite, unifies the two cases.)

**Orthogonal Projection** Truncating the orthonormal expansion (1.85a) to  $k \in \mathcal{I}$ , where  $\mathcal{I}$  is an index set that is a subset of the full index set  $\mathcal{K}$ ,  $\mathcal{I} \subset \mathcal{K}$ , gives an orthogonal projection. As in the alternate proof of Theorem 1.39, this can be

## 1.5. Bases and Frames

75

verified through somewhat tedious manipulations of sums and inner products; it is simpler to extend the definitions of the synthesis and analysis operators to apply for  $\mathcal{I} \subset \mathcal{K}$  and then use these new operators.

Define the *synthesis operator* associated with  $\{\varphi_k\}_{k \in \mathcal{I}}$  as

$$\Phi_{\mathcal{I}} : \ell^2(\mathcal{I}) \rightarrow H, \quad \text{with} \quad \Phi_{\mathcal{I}} \alpha = \sum_{k \in \mathcal{I}} \alpha_k \varphi_k. \quad (1.94)$$

This follows the form of (1.81) exactly, but the subscript  $\mathcal{I}$  emphasizes that  $\{\varphi_k\}_{k \in \mathcal{I}}$  may not be a basis. The adjoint of  $\Phi_{\mathcal{I}}$ , called the *analysis operator* associated with  $\{\varphi_k\}_{k \in \mathcal{I}}$ , is

$$\Phi_{\mathcal{I}}^* : H \rightarrow \ell^2(\mathcal{I}), \quad \text{with} \quad (\Phi_{\mathcal{I}}^* x)_k = \langle x, \varphi_k \rangle, \quad k \in \mathcal{I}. \quad (1.95)$$

Following the same steps as in (1.93), the orthonormality of the set  $\{\varphi_k\}_{k \in \mathcal{I}}$  is equivalent to

$$\Phi_{\mathcal{I}}^* \Phi_{\mathcal{I}} = I \quad \text{on} \quad \ell^2(\mathcal{I}). \quad (1.96)$$

However, we cannot conclude that  $\Phi_{\mathcal{I}}$  and  $\Phi_{\mathcal{I}}^*$  are unitary because the product  $\Phi_{\mathcal{I}} \Phi_{\mathcal{I}}^*$  is not, in general, the identity operator on  $H$ ;  $\Phi_{\mathcal{I}}$  not being a basis, it cannot reconstruct every  $x \in H$ . Instead,  $\Phi_{\mathcal{I}} \Phi_{\mathcal{I}}^*$  is an orthogonal projection operator that is an identity only when  $\mathcal{I} = \mathcal{K}$ , that is, when  $\{\varphi_k\}_{k \in \mathcal{I}}$  is a basis. This is formalized in the following theorem:

**THEOREM 1.40** Given an orthonormal set  $\Phi = \{\varphi_k\}_{k \in \mathcal{I}} \subset H$ ,

$$P_{\mathcal{I}} x = \sum_{k \in \mathcal{I}} \langle x, \varphi_k \rangle \varphi_k \quad (1.97a)$$

$$= \Phi_{\mathcal{I}} \Phi_{\mathcal{I}}^* x \quad (1.97b)$$

is the orthogonal projection of  $x$  onto  $S_{\mathcal{I}} = \overline{\text{span}}(\{\varphi_k\}_{k \in \mathcal{I}})$ .

*Proof.* From its definition (1.97a),  $P_{\mathcal{I}}$  is clearly a linear operator on  $H$  with range contained in  $S_{\mathcal{I}}$ . To prove that  $P_{\mathcal{I}}$  is an orthogonal projection operator, we show that it is idempotent and self-adjoint (see Definition 1.27).

The operator  $P_{\mathcal{I}}$  is idempotent because, for any  $x \in H$ ,

$$\begin{aligned} P_{\mathcal{I}}(P_{\mathcal{I}} x) &\stackrel{(a)}{=} \Phi_{\mathcal{I}} \Phi_{\mathcal{I}}^* (\Phi_{\mathcal{I}} \Phi_{\mathcal{I}}^* x) \stackrel{(b)}{=} \Phi_{\mathcal{I}} (\Phi_{\mathcal{I}}^* \Phi_{\mathcal{I}}) \Phi_{\mathcal{I}}^* x \\ &\stackrel{(c)}{=} \Phi_{\mathcal{I}} \Phi_{\mathcal{I}}^* x \stackrel{(d)}{=} P_{\mathcal{I}} x, \end{aligned}$$

where (a) follows from (1.97b); (b) from associativity of linear operators; (c) from (1.96); and (d) from (1.97b). This shows that  $P_{\mathcal{I}}$  is a projection operator. The operator  $P_{\mathcal{I}}$  is self-adjoint because

$$P_{\mathcal{I}}^* = (\Phi_{\mathcal{I}} \Phi_{\mathcal{I}}^*)^* \stackrel{(a)}{=} (\Phi_{\mathcal{I}}^*)^* \Phi_{\mathcal{I}}^* \stackrel{(b)}{=} \Phi_{\mathcal{I}} \Phi_{\mathcal{I}}^* = P_{\mathcal{I}},$$

where (a) follows from Theorem 1.21(viii); and (b) from Theorem 1.21(iii). Combined with the previous computation, this shows that  $P_{\mathcal{I}}$  is an orthogonal projection operator.

The previous theorem can be used to simplify the computation of an orthogonal projection provided  $\{\varphi_k\}_{k \in \mathcal{I}}$  is an orthonormal basis for the subspace of interest.

**EXAMPLE 1.33 (ORTHOGONAL PROJECTION WITH ORTHONORMAL BASIS)** Consider the orthonormal basis for  $\mathbb{C}^3$  from Example 1.30. The 2-dimensional subspace

$$S = \left\{ \begin{bmatrix} x_0 & x_1 & x_2 \end{bmatrix}^T \in \mathbb{C}^3 \mid x_1 = x_0 + x_2 \right\}$$

is  $\text{span}(\{\varphi_0, \varphi_1\})$ . Therefore, using (1.97a), the orthogonal projection onto  $S$  is given by

$$P_S x = \sum_{k=0}^1 \langle x, \varphi_k \rangle \varphi_k.$$

To see explicitly that this is an orthogonal projection operator:

$$\begin{aligned} P_S x &= \langle x, \varphi_0 \rangle \varphi_0 + \langle x, \varphi_1 \rangle \varphi_1 \stackrel{(a)}{=} \varphi_0 \langle x, \varphi_0 \rangle + \varphi_1 \langle x, \varphi_1 \rangle \\ &\stackrel{(b)}{=} \varphi_0 \varphi_0^* x + \varphi_1 \varphi_1^* x \stackrel{(c)}{=} (\varphi_0 \varphi_0^* + \varphi_1 \varphi_1^*) x \\ &= \frac{1}{3} \begin{bmatrix} 2 & 1 & -1 \\ 1 & 2 & 1 \\ -1 & 1 & 2 \end{bmatrix} x, \end{aligned}$$

where (a) follows from the inner product being a scalar; (b) from writing the inner product as a product of a row vector and a column vector; and (c) from the distributive property of matrix–vector multiplication. The matrix representation of  $P_S$  is idempotent (as can be verified with a straightforward computation) and obviously Hermitian.

**Orthogonal Decomposition** Given the orthogonal projection interpretation from Theorem 1.40, term  $k$  in the orthonormal expansion formula (1.85a) is the orthogonal projection of  $x$  to the subspace  $S_{\{k\}} = \text{span}(\varphi_k)$  (recall also Example 1.19). So (1.85a) writes any  $x$  uniquely as a sum of its orthogonal projections onto orthogonal 1-dimensional subspaces  $\{S_{\{k\}}\}_{k \in \mathcal{K}}$ . In other words, an orthonormal basis induces an orthogonal decomposition

$$H = \bigoplus_{k \in \mathcal{K}} S_{\{k\}} \quad (1.98)$$

while providing a simple way to compute the components of the decomposition of any  $x \in H$ . The expansion formula (1.85a) will be applied countless times in subsequent chapters, and orthogonal decompositions of Hilbert spaces  $\ell^2(\mathbb{Z})$  and  $\mathcal{L}^2(\mathbb{R})$  will be a recurring theme.

**Best Approximation** The simple form of (1.97a) makes certain sequences of orthogonal projections extremely easy to compute. Let  $\hat{x}^{(k)}$  denote the best approximation of  $x$  in the subspace spanned by the orthonormal set  $\{\varphi_0, \varphi_1, \dots, \varphi_{k-1}\}$ . Then  $\hat{x}^{(0)} = \mathbf{0}$  and

$$\hat{x}^{(k+1)} = \hat{x}^{(k)} + \langle x, \varphi_k \rangle \varphi_k \quad \text{for } k = 0, 1, \dots, \quad (1.99)$$

that is, the new best approximation is the sum of the previous best approximation plus the orthogonal projection onto the span of the added vector  $\varphi_k$ . This follows from the projection theorem (Theorem 1.26) and comparing (1.97a) with index sets  $\{0, 1, \dots, k-1\}$  and  $\{0, 1, \dots, k\}$ .

The recursive computation (1.99) is called *successive approximation*; it arises from the interest in nested subspaces and having orthonormal bases for those subspaces. Nested subspaces arise in practice quite frequently. For example, suppose we wish to find an approximation of a function  $x$  by a polynomial of minimal degree that meets an approximation error criterion. Then if  $\{\varphi_k\}_{k \in \mathbb{N}}$  is an orthonormal set such that, for each  $M$ ,  $\{\varphi_0, \varphi_1, \dots, \varphi_M\}$  is a basis for degree- $M$  polynomials, we can apply the recursion (1.99) until the error criterion is met. Gram–Schmidt orthogonalization, discussed below, is a way to find the desired set  $\{\varphi_k\}_{k \in \mathbb{N}}$ , and approximation by polynomials is covered in detail in Section 5.2.

**Bessel's Inequality** While Bessel's inequality is similar to Parseval's equality (1.87a), it holds for any orthonormal set—even if that set is not a basis. When it holds with equality (for *all* vectors in a Hilbert space, giving Parseval's equality), the orthonormal set must be a basis.

**THEOREM 1.41 (BESSEL'S INEQUALITY)** Given an orthonormal set  $\Phi = \{\varphi_k\}_{k \in \mathcal{I}}$  in a Hilbert space  $H$ , *Bessel's inequality* holds:

$$\|x\|^2 \geq \sum_{k \in \mathcal{I}} |\langle x, \varphi_k \rangle|^2 \quad (1.100a)$$

$$= \|\Phi_{\mathcal{I}}^* x\|^2. \quad (1.100b)$$

Equality for every  $x$  in  $H$  implies that the set  $\Phi$  is complete in  $H$ , so the orthonormal set is an orthonormal basis for  $H$ ; (1.100) is then Parseval's equality (1.87).

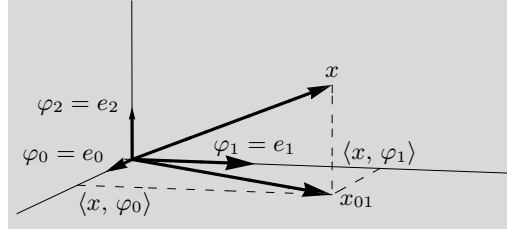
*Proof.* Let  $S = \overline{\text{span}}(\Phi)$  and  $x_S = \Phi_{\mathcal{I}} \Phi_{\mathcal{I}}^* x$ . By (1.97b),  $x_S$  is the orthogonal projection of  $x$  onto  $S$ . Thus, by the projection theorem (Theorem 1.26),  $x - x_S \perp x_S$ . From this we conclude

$$\|x\|^2 \stackrel{(a)}{=} \|x_S\|^2 + \|x - x_S\|^2 \stackrel{(b)}{\geq} \|x_S\|^2 = \|\Phi_{\mathcal{I}} \Phi_{\mathcal{I}}^* x\|^2 \stackrel{(c)}{=} \|\Phi_{\mathcal{I}}^* x\|^2 \stackrel{(d)}{=} \sum_{k \in \mathcal{I}} |\langle x, \varphi_k \rangle|^2,$$

where (a) follows from the Pythagorean theorem (1.26a); (b) from the nonnegativity of the norm of a vector; (c) from (1.96); and (d) from the definition of the analysis operator, (1.95).

Step (b) holds with equality for every  $x$  in  $H$  if and only if  $x = x_S$  for every  $x$  in  $H$ . This occurs if and only if  $S = H$ , in which case we have that the set  $\Phi$  is complete and thus an orthonormal basis for  $H$ .

For the case when the orthonormal set  $\{\varphi_k\}_{k \in \mathcal{I}}$  is not complete, Bessel's inequality is especially easy to understand by extending the set to an orthonormal basis  $\{\varphi_k\}_{k \in \mathcal{K}}$



**Figure 1.20:** Illustration of Bessel's inequality in  $\mathbb{R}^3$ .

with  $\mathcal{K} \supset \mathcal{I}$ . Then Bessel's inequality follows from Parseval's equality because  $\sum_{k \in \mathcal{I}} |\langle x, \varphi_k \rangle|^2$  simply omits some nonnegative terms from  $\sum_{k \in \mathcal{K}} |\langle x, \varphi_k \rangle|^2$ . The following example illustrates this in  $\mathbb{R}^3$ .

**EXAMPLE 1.34 (BESSEL'S INEQUALITY)** Let  $\varphi_0 = [1 \ 0 \ 0]^T$  and  $\varphi_1 = [0 \ 1 \ 0]^T$ . These vectors are the first two elements of the standard basis in  $\mathbb{R}^3$ , and they are orthonormal. As illustrated in Figure 1.20, the norm of a vector  $x \in \mathbb{R}^3$  is at least as large as the norm of its projection onto the  $(\varphi_0, \varphi_1)$ -plane,  $x_{01}$ :

$$\|x\|^2 \geq \|x_{01}\|^2 = |\langle x, \varphi_0 \rangle|^2 + |\langle x, \varphi_1 \rangle|^2.$$

Adding  $\varphi_2 = [0 \ 0 \ 1]^T$  to the set gives an orthonormal basis (the standard basis), and adding the square of the length of the orthogonal projection of  $x$  onto the span of  $\varphi_2$  yields Parseval's equality:

$$\|x\|^2 = \sum_{k=0}^2 |\langle x, \varphi_k \rangle|^2.$$

**Gram–Schmidt Orthogonalization** We have thus far discussed properties of orthonormal bases and checked whether a set is an orthonormal basis. We now show how to construct an orthonormal basis for a space specified by a set of linearly independent vectors  $\{x_k\}_{k \in \mathcal{K}} \subset H$ . For notational convenience, assume  $\mathcal{K}$  is a set of consecutive integers starting at 0, so  $\mathcal{K} = \{0, 1, \dots, N-1\}$  or  $\mathcal{K} = \mathbb{N}$ .

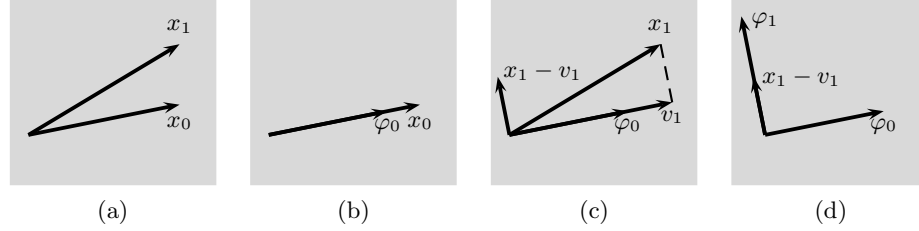
The goal is to find an orthonormal set  $\{\varphi_k\}_{k \in \mathcal{K}}$  with

$$\overline{\text{span}}(\{\varphi_k\}_{k \in \mathcal{K}}) = \overline{\text{span}}(\{x_k\}_{k \in \mathcal{K}}). \quad (1.101a)$$

Thus, when  $\{x_k\}_{k \in \mathcal{K}}$  is a basis for  $H$ , the constructed set  $\{\varphi_k\}_{k \in \mathcal{K}}$  is an orthonormal basis for  $H$ ; otherwise,  $\{\varphi_k\}_{k \in \mathcal{K}}$  is an orthonormal basis for the smaller space  $\overline{\text{span}}(\{x_k\}_{k \in \mathcal{K}})$ , which is itself a Hilbert space.

There are many orthonormal bases for  $\overline{\text{span}}(\{x_k\}_{k \in \mathcal{K}})$ . By requiring a stronger condition

$$\text{span}(\{\varphi_k\}_{k=0}^i) = \text{span}(\{x_k\}_{k=0}^i) \quad \text{for every } i \in \mathbb{N}, \quad (1.101b)$$



**Figure 1.21:** Illustration of Gram–Schmidt orthogonalization. (a) Input vectors  $\{x_0, x_1\}$ . (b) The first output vector  $\varphi_0$  is a normalized version of  $x_0$ . (c) The projection of  $x_1$  onto the subspace spanned by  $\varphi_0$  is subtracted from  $x_1$  to obtain a residual  $x_1 - v_1$ . (d) The second output vector  $\varphi_1$  is a normalized version of the residual.

the solution becomes essentially unique. Furthermore, enforcing (1.101b) for increasing values of  $i$  leads to a simple recursive procedure. Figure 1.21 illustrates the orthogonalization procedure for two vectors in a plane (initial, nonorthonormal basis). For example, for  $i = 0$ , (1.101b) holds when  $\varphi_0$  is a scalar multiple of  $x_0$ . For  $\varphi_0$  to have unit norm, it is natural to choose

$$\varphi_0 = x_0 / \|x_0\|,$$

as illustrated in Figure 1.21(b), and the set of all possible solutions is obtained by including a unit-modulus scalar factor. Then for (1.101b) to hold for  $i = 1$ , the vector  $\varphi_1$  must be aligned with the component of  $x_1$  orthogonal to  $\varphi_0$ , as illustrated in Figure 1.21(c). This is achieved when  $\varphi_1$  is a scalar multiple of the residual of orthogonally projecting  $x_1$  to the subspace spanned by  $\varphi_0$ , as illustrated in Figure 1.21(d):

$$\varphi_1 = \frac{x_1 - \langle x_1, \varphi_0 \rangle \varphi_0}{\|x_1 - \langle x_1, \varphi_0 \rangle \varphi_0\|}.$$

In general,  $\varphi_k$  is determined by normalizing the residual of orthogonally projecting  $x_k$  to  $\text{span}(\{\varphi_0, \varphi_1, \dots, \varphi_{k-1}\})$ . The residual is nonzero because otherwise the linear independence of  $\{x_0, x_1, \dots, x_k\}$  is contradicted. The full recursive computation is summarized in Table 1.1.

#### EXAMPLE 1.35 (GRAM–SCHMIDT ORTHOGONALIZATION)

- (i) Let  $x_0 = [1 \ 1 \ 0]^T$ ,  $x_1 = [0 \ 1 \ 1]^T$ , and  $x_2 = [1 \ 1 \ 1]^T$ . These are linearly independent, and following the steps in Table 1.1 first yields  $\varphi_0 = \frac{1}{\sqrt{2}} [1 \ 1 \ 0]^T$ , then  $v_1 = \frac{1}{2} [1 \ 1 \ 0]^T$ , then  $x_1 - v_1 = \frac{1}{2} [-1 \ 1 \ 2]^T$ , and  $\varphi_1 = \frac{1}{\sqrt{6}} [-1 \ 1 \ 2]^T$ . For the final basis vector,  $v_2 = \frac{1}{3} [2 \ 4 \ 2]^T$ ,  $x_2 - v_2 = \frac{1}{3} [1 \ -1 \ 1]^T$ , and  $\varphi_2 = \frac{1}{\sqrt{3}} [1 \ -1 \ 1]^T$ . The set  $\{\varphi_0, \varphi_1, \varphi_2\}$  is the orthonormal basis from Examples 1.30 and 1.33. Since  $\text{span}(\{\varphi_0, \varphi_1\}) = \text{span}(\{x_0, x_1\})$  and the latter span is plainly the range of matrix  $B$  in Example 1.20, we can retrospectively see that the projection operators in

**Gram–Schmidt Orthogonalization****Input:** An ordered sequence of linearly independent vectors  $\{x_k\}_{k \in \mathcal{K}} \subset H$ **Output:** Orthonormal vectors  $\{\varphi_k\}_{k \in \mathcal{K}} \subset H$ , with  $\text{span}(\{\varphi_k\}) = \text{span}(\{x_k\})$ 


---

```

 $\{\varphi_k\} = \text{GramSchmidt}(\{x_k\})$ 
 $\varphi_0 = x_0 / \|x_0\|$ 
 $k = 1$ 
while  $\varphi_k$  exists do
    project  $v_k = \sum_{i=0}^{k-1} \langle x_k, \varphi_i \rangle \varphi_i$ 
    normalize  $\varphi_k = (x_k - v_k) / \|x_k - v_k\|$ 
    increment  $k$ 
end while
return  $\{\varphi_k\}$ 

```

---

**Table 1.1:** Gram–Schmidt orthogonalization algorithm.

Examples 1.20 and 1.33 project to the same subspace. (One projection operator is orthogonal and the other is oblique.)

- (ii) Starting with  $x_0 = [1 \ 1 \ 0]^T$  and  $x_1 = [0 \ 1 \ 1]^T$  would again yield  $\varphi_0 = \frac{1}{\sqrt{2}} [1 \ 1 \ 0]^T$  and  $\varphi_1 = \frac{1}{\sqrt{6}} [-1 \ 1 \ 2]^T$ . Now  $\{\varphi_0, \varphi_1\}$  is obviously too small to be a basis for  $\mathbb{C}^3$ . Instead, it is an orthonormal basis for the 2-dimensional space  $\text{span}(\{x_0, x_1\})$ . As discussed in Examples 1.20 and 1.33, this space is the set of 3-tuples with middle component equal to the sum of the first and last.
- (iii) Starting with  $x_0 = [1 \ 1 \ 1]^T$ ,  $x_1 = [1 \ 1 \ 0]^T$ , and  $x_2 = [0 \ 1 \ 1]^T$ , the same set of vectors as in (i), but in a different order, yields

$$\varphi_0 = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \quad \varphi_1 = \frac{1}{\sqrt{6}} \begin{bmatrix} 1 \\ 1 \\ -2 \end{bmatrix}, \quad \text{and} \quad \varphi_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}.$$

There is no obvious relationship between this orthonormal basis and the one found in part (i).

Solved Exercise 1.5 applies Gram–Schmidt orthogonalization to derive normalized Legendre polynomials, which are polynomials orthogonal in  $\mathcal{L}^2([-1, 1])$ .

**1.5.3 Biorthogonal Pairs of Bases**

Orthonormal bases have many advantages over other bases, including simple expressions for expansion in (1.85) and orthogonal projection in (1.97). While there are no general disadvantages caused directly by orthonormality, in some settings



nonorthonormal bases have their advantages, too. For example, of the bases

$$\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\} \quad \text{and} \quad \left\{ \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \frac{1}{\sqrt{6}} \begin{bmatrix} -1 \\ 1 \\ 2 \end{bmatrix}, \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} \right\}$$

from Example 1.35(i), the nonorthogonal basis is easier to store and compute with. Solved Exercise 1.5 provides a more dramatic example, since the set of functions  $\{1, t, t^2, \dots, t^N\}$  is certainly simpler for many purposes than the Legendre polynomials up to degree  $N$ .

A basis does not have to be orthonormal to provide unique expansions. The sacrifice we must make is that we cannot ask for a single set of vectors to serve the analysis role in  $x \mapsto \alpha = \{\langle x, \varphi_k \rangle\}_{k \in \mathcal{K}}$  and the synthesis role in  $\alpha \mapsto \sum_{k \in \mathcal{K}} \alpha_k \varphi_k$ . This leads us to the concept of a *biorthogonal pair of bases*, or *dual bases*.

**DEFINITION 1.42 (BIORTHOGONAL PAIR OF BASES)** The sets of vectors  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}} \subset H$  and  $\tilde{\Phi} = \{\tilde{\varphi}_k\}_{k \in \mathcal{K}} \subset H$ , where  $\mathcal{K}$  is finite or countably infinite, are called a biorthogonal pair of bases for Hilbert space  $H$  when

- (i) each is a *basis* for  $H$ ; and
- (ii) they are *biorthogonal*, meaning

$$\langle \varphi_i, \tilde{\varphi}_k \rangle = \delta_{i-k} \quad \text{for every } i, k \in \mathcal{K}. \quad (1.102)$$

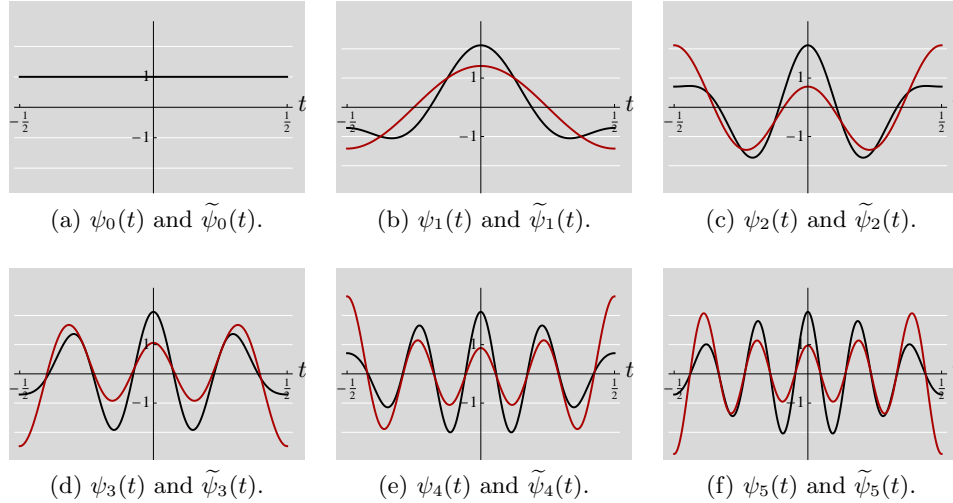
Since the inner product has Hermitian symmetry and  $\delta_{i-k}$  is real, the roles of the sets  $\Phi$  and  $\tilde{\Phi}$  can be reversed with no change in whether (1.102) holds. We will generally maintain a convention of using the basis  $\Phi$  in synthesis and the basis  $\tilde{\Phi}$  in analysis, with the understanding that the bases can be swapped in any of the results that follow. To each basis we associate synthesis and analysis operators defined through (1.81) and (1.82); a biorthogonal pair of bases thus yields four operators:  $\Phi$ ,  $\Phi^*$ ,  $\tilde{\Phi}$ , and  $\tilde{\Phi}^*$ .

**EXAMPLE 1.36 (FINITE-DIMENSIONAL BIORTHOGONAL PAIR OF BASES)** The sets

$$\varphi_0 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \varphi_1 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \varphi_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad \text{and} \quad \tilde{\varphi}_0 = \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix}, \tilde{\varphi}_1 = \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}, \tilde{\varphi}_2 = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$$

are a biorthogonal pair of bases for  $\mathbb{C}^3$ , as can be verified by direct computation.

**EXAMPLE 1.37 (BIORTHOGONAL PAIR OF BASES OF COSINE FUNCTIONS)** Define



**Figure 1.22:** Elements of the biorthogonal pair of bases  $\Psi$  (solid lines) and  $\tilde{\Psi}$  (dashed lines) in Example 1.37.

$\Psi = \{\psi_k\}_{k \in \mathbb{N}} \subset \mathcal{L}^2([-\frac{1}{2}, \frac{1}{2}])$  and  $\tilde{\Psi} = \{\tilde{\psi}_k\}_{k \in \mathbb{N}} \subset \mathcal{L}^2([-\frac{1}{2}, \frac{1}{2}])$  by

$$\begin{aligned} \psi_0(t) &= 1 \\ &= \varphi_0(t), \end{aligned} \tag{1.103a}$$

$$\begin{aligned} \psi_k(t) &= \sqrt{2} \cos(2\pi kt) + \frac{1}{2} \sqrt{2} \cos(2\pi(k+1)t) \\ &= \varphi_k(t) + \frac{1}{2} \varphi_{k+1}(t), \quad k = 1, 2, \dots, \end{aligned} \tag{1.103b}$$

$$\begin{aligned} \tilde{\psi}_0(t) &= 1 \\ &= \varphi_0(t), \end{aligned} \tag{1.103c}$$

$$\begin{aligned} \tilde{\psi}_k(t) &= \sum_{m=1}^k \left(-\frac{1}{2}\right)^{k-m} \sqrt{2} \cos(2\pi mt) \\ &= \sum_{m=1}^k \left(-\frac{1}{2}\right)^{k-m} \varphi_m(t), \quad k \in \mathbb{N}, \end{aligned} \tag{1.103d}$$

where  $\{\varphi_k\}_{k \in \mathbb{Z}}$  are the orthonormal basis functions from (1.21). The first few functions in each of these sets are shown in Figure 1.22. Verifying that (1.102) holds is only part of proving that the sets  $\Psi$  and  $\tilde{\Psi}$  form a biorthogonal pair of bases; this is left for Exercise 1.43. We must also verify that each set is a basis for the same subspace of  $\mathcal{L}^2([-\frac{1}{2}, \frac{1}{2}])$ .

By construction,  $\{\varphi_k\}_{k \in \mathbb{N}}$  forms an orthonormal basis for the closure of its span  $S$ . We will see that the sets  $\Psi$  and  $\tilde{\Psi}$  are also bases for  $S$ . The set  $\Psi$  is linearly independent because no function  $\psi_i$  can be written as a linear combination of  $\{\psi_k\}_{k=0}^{i-1}$ ; this follows from  $\psi_i$  containing a higher-frequency cosine

## 1.5. Bases and Frames

83

than any lower-numbered element from the set. The set  $\tilde{\Psi}$  is similarly linearly independent. The closure of the span of  $\Psi$  and  $S$  are equal:  $\overline{\text{span}}(\Psi) \subset S$  because each  $\psi_k$  is a linear combination of one or two  $\varphi_k$ s; and  $S \subset \overline{\text{span}}(\Psi)$  because  $\varphi_0 = \psi_0$  and for each  $k \in \mathbb{Z}^+$ ,  $\varphi_k$  can be written as an infinite linear combination of  $\psi_k$ s; a detailed argument is left for Exercise 1.43. Similarly, the closure of the span of the set  $\tilde{\Psi}$  is equal to  $S$ .

**Expansion and Inner Product Computation** With a biorthogonal pair of bases, expansion coefficients with respect to one basis are computed using the other basis.

**THEOREM 1.43 (BIORTHOGONAL BASIS EXPANSIONS)** Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}}$  and  $\tilde{\Phi} = \{\tilde{\varphi}_k\}_{k \in \mathcal{K}}$  be a biorthogonal pair of bases for Hilbert space  $H$ . The unique expansion with respect to the basis  $\Phi$  of any  $x$  in  $H$  has expansion coefficients

$$\alpha_k = \langle x, \tilde{\varphi}_k \rangle \quad \text{for } k \in \mathcal{K}, \quad \text{or,} \quad (1.104a)$$

$$\alpha = \tilde{\Phi}^* x. \quad (1.104b)$$

Synthesis with these coefficients yields

$$x = \sum_{k \in \mathcal{K}} \langle x, \tilde{\varphi}_k \rangle \varphi_k \quad (1.105a)$$

$$= \Phi \alpha = \Phi \tilde{\Phi}^* x. \quad (1.105b)$$

*Proof.* The proof parallels the proof of Theorem 1.38 with minor modifications based on replacing orthonormality condition (1.83) with biorthogonality condition (1.102).

The existence of a unique linear combination of the form (1.79) is guaranteed by the set  $\Phi$  being a basis. The validity of (1.105a) with coefficients (1.104a) follows from the following computation:

$$\langle x, \tilde{\varphi}_k \rangle \stackrel{(a)}{=} \left\langle \sum_{i \in \mathcal{K}} \alpha_i \varphi_i, \tilde{\varphi}_k \right\rangle \stackrel{(b)}{=} \sum_{i \in \mathcal{K}} \alpha_i \langle \varphi_i, \tilde{\varphi}_k \rangle \stackrel{(c)}{=} \sum_{i \in \mathcal{K}} \alpha_i \delta_{i-k} \stackrel{(d)}{=} \alpha_k,$$

where (a) follows from (1.79); (b) from the linearity in the first argument of the inner product; (c) from the biorthogonality of the sets  $\Phi$  and  $\tilde{\Phi}$ , (1.102); and (d) from the definition of the Kronecker delta sequence, (1.9).

The expressions (1.104b) and (1.105b) are equivalent to (1.104a) and (1.105a) using the operators defined in (1.81) and (1.82).

Reversing the roles of the bases  $\Phi$  and  $\tilde{\Phi}$  gives expansion coefficients with respect to the basis  $\tilde{\Phi}$ :

$$\tilde{\alpha}_k = \langle x, \varphi_k \rangle \quad \text{for } k \in \mathcal{K}, \quad \text{or,} \quad \tilde{\alpha} = \Phi^* x, \quad (1.106)$$

with the corresponding expansion

$$x = \sum_{k \in \mathcal{K}} \langle x, \varphi_k \rangle \tilde{\varphi}_k = \tilde{\Phi} \Phi^* x. \quad (1.107)$$

The theorem shows that a biorthogonal pair of bases can together do the job of an orthonormal basis in terms of signal expansion. The most interesting properties of the synthesis and analysis operators involve both bases of the pair. Since (1.105b) holds for all  $x$  in  $H$ ,

$$\Phi \tilde{\Phi}^* = I \quad \text{on } H. \quad (1.108)$$

This leads to an analogue of Theorem 1.39:

**THEOREM 1.44 (PARSEVAL'S EQUALITIES FOR BIORTHOGONAL PAIRS OF BASES)**  
Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}}$  and  $\tilde{\Phi} = \{\tilde{\varphi}_k\}_{k \in \mathcal{K}}$  be a biorthogonal pair of bases for Hilbert space  $H$ . Expansion with respect to the bases  $\Phi$  and  $\tilde{\Phi}$  with coefficients (1.104) and (1.106) satisfies

$$\|x\|^2 = \sum_{k \in \mathcal{K}} \langle x, \varphi_k \rangle \langle x, \tilde{\varphi}_k \rangle^* \quad (1.109a)$$

$$= \langle \Phi^* x, \tilde{\Phi}^* x \rangle = \langle \tilde{\alpha}, \alpha \rangle. \quad (1.109b)$$

More generally,

$$\langle x, y \rangle = \sum_{k \in \mathcal{K}} \langle x, \varphi_k \rangle \langle y, \tilde{\varphi}_k \rangle^* \quad (1.110a)$$

$$= \langle \Phi^* x, \tilde{\Phi}^* y \rangle = \langle \tilde{\alpha}, \beta \rangle. \quad (1.110b)$$

*Proof.* We will prove (1.110b); (1.110a) is the same fact expanded with the definitions of  $\Phi^*$  and  $\tilde{\Phi}^*$ , and equalities (1.109) follow by setting  $x = y$ . For any  $x$  and  $y$  in  $H$ ,

$$\langle \Phi^* x, \tilde{\Phi}^* y \rangle \stackrel{(a)}{=} \langle x, \Phi \tilde{\Phi}^* y \rangle \stackrel{(b)}{=} \langle x, y \rangle,$$

where (a) follows from the definition of adjoint; and (b) from (1.108).

**Gram Matrix** Theorem 1.44 is not nearly as useful as Theorem 1.39 because it involves expansions with respect to both bases of the pair. In (1.110),  $x = \sum_{k \in \mathcal{K}} \tilde{\alpha}_k \tilde{\varphi}_k$  and  $y = \sum_{k \in \mathcal{K}} \beta_k \varphi_k$  (note the use of different bases) so that

$$\langle x, y \rangle = \langle \tilde{\alpha}, \beta \rangle = \sum_{k \in \mathcal{K}} \tilde{\alpha}_k \beta_k^*.$$

More often, one wants all expansions to be with respect to one basis of the pair; the other basis of the pair serves as a helper in computing the expansion coefficients. If  $x = \Phi \alpha$  and  $y = \Phi \beta$  (both expansions with respect to the basis  $\Phi$ ), then

$$\langle x, y \rangle = \langle \Phi \alpha, \Phi \beta \rangle \stackrel{(a)}{=} \langle \Phi^* \Phi \alpha, \beta \rangle \stackrel{(b)}{=} \langle G \alpha, \beta \rangle, \quad (1.111)$$

## 1.5. Bases and Frames

85

where (a) follows from the meaning of adjoint; and (b) from introducing the *Gram matrix* or *Gramian*  $G$ ,

$$G = \Phi^* \Phi, \quad (1.112a)$$

$$G_{ik} = \langle \varphi_k, \varphi_i \rangle \quad \text{for every } i, k \in \mathcal{K}, \quad (1.112b)$$

$$G = \begin{bmatrix} \vdots & \vdots & \vdots \\ \cdots & \langle \varphi_{-1}, \varphi_{-1} \rangle & \langle \varphi_0, \varphi_{-1} \rangle & \langle \varphi_1, \varphi_{-1} \rangle & \cdots \\ \cdots & \langle \varphi_{-1}, \varphi_0 \rangle & \boxed{\langle \varphi_0, \varphi_0 \rangle} & \langle \varphi_1, \varphi_0 \rangle & \cdots \\ \cdots & \langle \varphi_{-1}, \varphi_1 \rangle & \langle \varphi_0, \varphi_1 \rangle & \langle \varphi_1, \varphi_1 \rangle & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix}. \quad (1.112c)$$

The order of factors in (1.111) evokes a product of three matrices:

$$\langle x, y \rangle = \beta^* G \alpha, \quad (1.113)$$

where as before  $\alpha$  and  $\beta$  are column vectors. When the set  $\Phi$  is an orthonormal basis,  $G$  simplifies to the identity operator on  $\ell^2(\mathcal{K})$  and (1.113) simplifies to (1.90).

**EXAMPLE 1.38 ( $\mathbb{C}^3$  INNER PRODUCT COMPUTATION WITH BASES)** Consider the basis  $\{\varphi_0, \varphi_1, \varphi_2\} \subset \mathbb{C}^3$  from Example 1.36. The Gram matrix of this basis is

$$G = \Phi^* \Phi = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 2 \\ 1 & 2 & 2 \\ 2 & 2 & 3 \end{bmatrix}.$$

For any  $x$  and  $y$  in  $\mathbb{C}^3$ , the expansions with respect to the basis  $\Phi$  are  $\alpha = \tilde{\Phi}^* x$  and  $\beta = \tilde{\Phi}^* y$ , where

$$\tilde{\Phi}^* = \begin{bmatrix} 0 & 1 & -1 \\ -1 & 1 & 0 \\ 1 & -1 & 1 \end{bmatrix}$$

is the analysis operator associated with the basis  $\{\tilde{\varphi}_0, \tilde{\varphi}_1, \tilde{\varphi}_2\} \subset \mathbb{C}^3$  from Example 1.36. Then  $\langle x, y \rangle = \beta^* G \alpha$  by using (1.113).

In  $\mathbb{C}^N$ , it is often natural and easy to use the standard basis for inner product computations. Thus, the previous example may seem to be a complicated way to achieve a simple result. In fact, we have

$$\beta^* G \alpha = (\tilde{\Phi}^* y)^* (\Phi^* \Phi) (\tilde{\Phi}^* x) = y^* \tilde{\Phi} \Phi^* \Phi \tilde{\Phi}^* x = y^* (\Phi \tilde{\Phi}^*)^* (\Phi \tilde{\Phi}^*) x,$$

so in light of (1.108), the inner product  $y^* x$  has been altered only by the insertion of identity operators. The use of (1.113) is more valuable when expansion with respect to a biorthogonal basis is natural and precomputation of  $G$  avoids a laborious inner product computation, as we illustrate next.

**EXAMPLE 1.39 (POLYNOMIAL INNER PRODUCT COMPUTATION WITH BASES)** Consider the polynomials of degree at most 3 under the  $\mathcal{L}^2([-1, 1])$  inner product. The basis  $\{1, t, t^2, t^3\}$  is easy to use because the expansion coefficients

of a polynomial  $x(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2 + \alpha_3 t^3$  are read off directly as  $\alpha = [\alpha_0 \ \alpha_1 \ \alpha_2 \ \alpha_3]^T$ . However, since the basis is not orthonormal, we cannot compute inner products with (1.90). Instead, since

$$\langle t^k, t^i \rangle = \int_{-1}^1 t^k t^i dt = \frac{1}{i+k+1} (1 - (-1)^{i+k+1}),$$

the Gram matrix of the basis is

$$G = \begin{bmatrix} 2 & 0 & 2/3 & 0 \\ 0 & 2/3 & 0 & 2/5 \\ 2/3 & 0 & 2/5 & 0 \\ 0 & 2/5 & 0 & 2/7 \end{bmatrix},$$

and inner products can be computed without any integration using (1.113).

**Inverse Synthesis and Analysis** Equation (1.108) shows that  $\tilde{\Phi}^*$  is a right inverse of  $\Phi$ . It is also true that

$$\tilde{\Phi}^* \Phi = I \quad \text{on } \ell^2(\mathcal{K}), \quad (1.114)$$

making  $\tilde{\Phi}^*$  a left inverse of  $\Phi$  and furthermore showing that  $\tilde{\Phi}^*$  is the unique inverse of  $\Phi$ . To verify (1.114), make the following computation for any sequence  $\alpha$  in  $\ell^2(\mathcal{K})$ :

$$\begin{aligned} \tilde{\Phi}^* \Phi \alpha &\stackrel{(a)}{=} \tilde{\Phi}^* \sum_{i \in \mathcal{K}} \alpha_i \varphi_i \stackrel{(b)}{=} \{ \langle \sum_{i \in \mathcal{K}} \alpha_i \varphi_i, \tilde{\varphi}_k \rangle \}_{k \in \mathcal{K}} \stackrel{(c)}{=} \{ \sum_{i \in \mathcal{K}} \alpha_i \langle \varphi_i, \tilde{\varphi}_k \rangle \}_{k \in \mathcal{K}} \\ &\stackrel{(d)}{=} \{ \sum_{i \in \mathcal{K}} \alpha_i \delta_{i-k} \}_{k \in \mathcal{K}} \stackrel{(e)}{=} \{ \alpha_k \}_{k \in \mathcal{K}} = \alpha, \end{aligned} \quad (1.115)$$

where (a) follows from (1.81); (b) from (1.82); (c) from the linearity in the first argument of the inner product; (d) from biorthogonality of the sets  $\{\varphi_k\}_{k \in \mathcal{K}}$  and  $\{\tilde{\varphi}_k\}_{k \in \mathcal{K}}$ , (1.102); and (e) from the definition of the Kronecker delta sequence, (1.9).

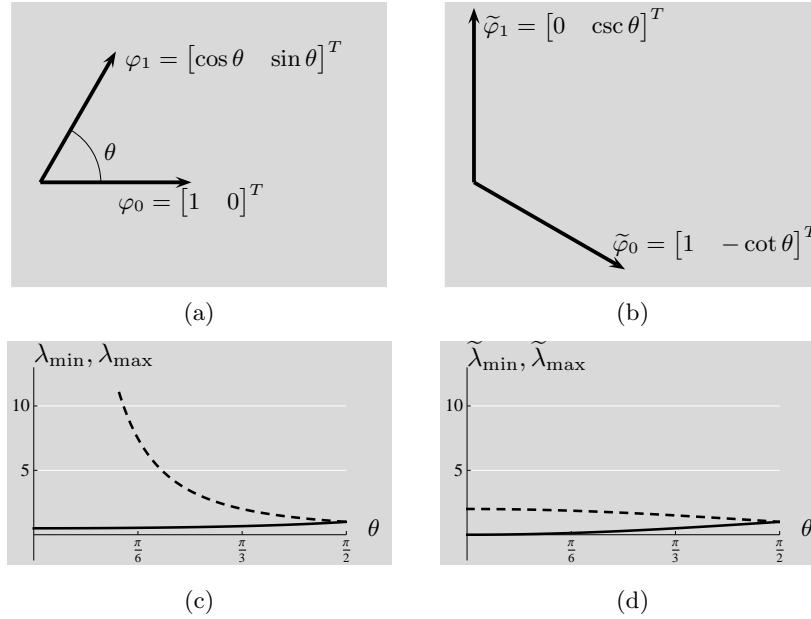
Knowing that operators associated with a biorthogonal pair of bases satisfy

$$\tilde{\Phi}^* = \Phi^{-1} \quad (1.116)$$

can be used to determine  $\tilde{\Phi}$  from  $\Phi$  such that the sets  $\Phi$  and  $\tilde{\Phi}$  form a biorthogonal pair of bases. A simple special case is when Hilbert space  $H$  is  $\mathbb{C}^N$  (or  $\mathbb{R}^N$ ). Then, synthesis operator  $\Phi$  is an  $N \times N$  matrix with the basis vectors as columns, and linear independence of the basis implies that  $\Phi$  is invertible. Setting  $\tilde{\Phi} = (\Phi^{-1})^*$  means that the vectors of the dual basis are the conjugate transposes of the rows of  $\Phi^{-1}$ . It is a valuable exercise to check that in Example 1.36,  $\tilde{\Phi}$  can be seen as derived from  $\Phi$  in this manner.

**EXAMPLE 1.40 (DUAL BASES, EXAMPLE 1.28 CONT'D)** Take the basis  $\Phi$  from Example 1.28(ii). Assuming  $\theta \neq 0$ , and using (1.116), the basis  $\Phi$  and its dual basis  $\tilde{\Phi}$  are

$$\Phi = \begin{bmatrix} 1 & \cos \theta \\ 0 & \sin \theta \end{bmatrix}, \quad \tilde{\Phi} = \begin{bmatrix} 1 & 0 \\ -\cot \theta & \csc \theta \end{bmatrix}, \quad (1.117)$$



**Figure 1.23:** A biorthogonal pair of bases in  $\mathbb{R}^2$  and their Riesz basis stability constants. (a) Basis  $\Phi$  from Figure 1.18(b). (b) Its corresponding dual basis  $\tilde{\Phi}$ . (c)  $\lambda_{\min}$  and  $\lambda_{\max}$  for the basis in (a) as a function of  $\theta$ . (d)  $\tilde{\lambda}_{\min}$  and  $\tilde{\lambda}_{\max}$  for the dual basis in (b) as a function of  $\theta$ . The Riesz basis constants here are the reciprocals of the ones in (c).

both shown in Figure 1.23(a) and (c). We can easily check that the biorthogonality condition (1.102) holds. The figure also illustrates how a unit-norm basis does not necessarily lead to a unit-norm dual basis.

We have already computed  $\lambda_{\min}$  and  $\lambda_{\max}$  for the basis in (a); we can similarly find that for the dual basis,

$$\tilde{\lambda}_{\min} = \frac{1}{\lambda_{\max}}, \quad \tilde{\lambda}_{\max} = \frac{1}{\lambda_{\min}}, \quad (1.118)$$

shown in Figure 1.23(d). Clearly, the pair is best behaved for  $\theta = \pi/2$ , when it reduces to an orthonormal basis. As  $\theta$  approaches 0, the basis vectors in  $\Phi$  become close to colinear, destroying the basis property.

When the Hilbert space is not  $\mathbb{C}^N$  (or  $\mathbb{R}^N$ ), the simplicity of the equation  $\tilde{\Phi}^* = \Phi^{-1}$  is deceptive. The operators  $\tilde{\Phi}^*$  and  $\Phi^{-1}$  are mappings from  $H$  to  $\ell^2(\mathcal{K})$ . The analysis operator  $\tilde{\Phi}^*$  maps from  $H$  to  $\ell^2(\mathcal{K})$  through the inner products with  $\{\tilde{\varphi}_k\}_{k \in \mathcal{K}}$ . To determine  $\{\tilde{\varphi}_k\}_{k \in \mathcal{K}}$  from  $\Phi^{-1}$  is to interpret the operation of  $\Phi^{-1}$  as computing inner products with some set of vectors. We derive that set of vectors next.

**Dual Basis** So far we have derived properties of a biorthogonal pair of bases. Now we show how to find the unique basis  $\tilde{\Phi}$  that completes a biorthogonal pair with a given Riesz basis  $\Phi$ . As noted above, when  $\Phi$  is a basis<sup>19</sup> for  $\mathbb{C}^N$ , finding the appropriate  $\tilde{\Phi}$  is as simple as inverting the matrix  $\Phi$ . In general, we find  $\tilde{\Phi}$  by imposing two key properties:  $\Phi$  and  $\tilde{\Phi}$  span the same space  $H$ , and the sets are biorthogonal.

Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}} \subset H$  be a Riesz basis for Hilbert space  $H$ . To ensure  $\overline{\text{span}}(\tilde{\Phi}) \subseteq \overline{\text{span}}(\Phi)$ , let

$$\tilde{\varphi}_k = \sum_{\ell \in \mathcal{K}} a_{\ell,k} \varphi_\ell, \quad \text{for each } k \in \mathcal{K}. \quad (1.119a)$$

This set of equations can be combined into a single matrix product equation to express the synthesis operator  $\tilde{\Phi}$  as

$$\tilde{\Phi} = \Phi A, \quad (1.119b)$$

where the  $(\ell, k)$  entry of  $A : \ell^2(\mathcal{K}) \rightarrow \ell^2(\mathcal{K})$  is  $a_{\ell,k}$ . Determining the coefficients  $a_{\ell,k}$  for  $k, \ell \in \mathcal{K}$  specifies the dual basis  $\tilde{\Phi}$  through either of the forms of (1.119).

The biorthogonality condition (1.102) dictates that for every  $i, k \in \mathcal{K}$ ,

$$\delta_{i-k} = \langle \varphi_i, \tilde{\varphi}_k \rangle \stackrel{(a)}{=} \langle \varphi_i, \sum_{\ell \in \mathcal{K}} a_{\ell,k} \varphi_\ell \rangle \stackrel{(b)}{=} \sum_{\ell \in \mathcal{K}} a_{\ell,k}^* \langle \varphi_i, \varphi_\ell \rangle \stackrel{(c)}{=} \sum_{\ell \in \mathcal{K}} a_{\ell,k}^* G_{\ell,i}, \quad (1.120)$$

where (a) uses (1.119a) to substitute for  $\tilde{\varphi}_k$ ; (b) follows from conjugate linearity in the second argument of the inner product; and (c) uses the Gram matrix defined in (1.112b). By taking the conjugate of both sides of (1.120) and using the Hermitian symmetry of the Gram matrix, we obtain

$$\delta_{i-k} = \sum_{\ell \in \mathcal{K}} G_{i,\ell} a_{\ell,k}, \quad \text{for every } i, k \in \mathcal{K}. \quad (1.121a)$$

This set of equations can be combined into a single matrix product equation

$$I = GA. \quad (1.121b)$$

Thus the inverse of the Gram matrix gives the desired coefficients.<sup>20</sup>

This derivation is summarized by the following theorem:

**THEOREM 1.45 (DUAL BASIS)** Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}}$  be a Riesz basis for Hilbert space  $H$ , and let  $A : \ell^2(\mathcal{K}) \rightarrow \ell^2(\mathcal{K})$  be the inverse of the Gram matrix of  $\Phi$ , that is,  $A = (\Phi^* \Phi)^{-1}$ . Then the set  $\tilde{\Phi} = \{\tilde{\varphi}_k\}_{k \in \mathcal{K}}$  defined via

$$\tilde{\varphi}_k = \sum_{\ell \in \mathcal{K}} a_{\ell,k} \varphi_\ell, \quad \text{for each } k \in \mathcal{K}, \quad (1.122a)$$

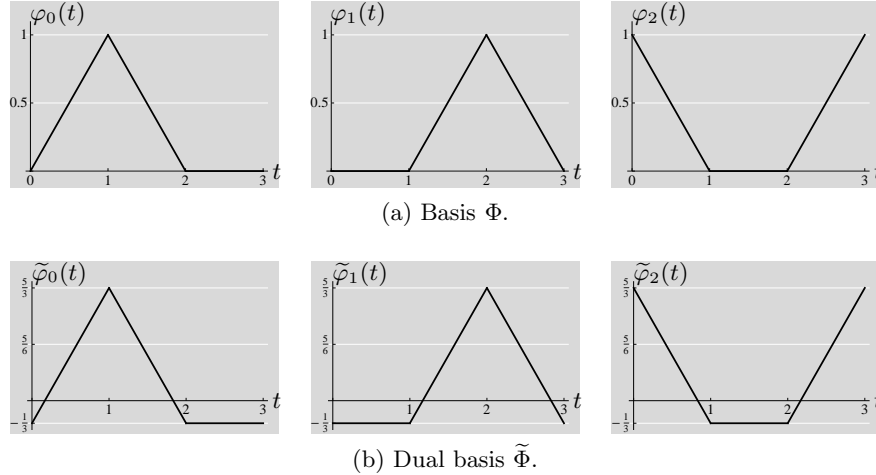
<sup>19</sup>Recall that in any finite-dimensional spaces, any basis is a Riesz basis.

<sup>20</sup>The Riesz basis condition on  $\Phi$  ensures that the inverse exists.



## 1.5. Bases and Frames

89



**Figure 1.24:** (a) Three functions  $\varphi_0$ ,  $\varphi_1$ , and  $\varphi_2$  related by circular shift form a basis  $\Phi$ . (b) The dual set  $\tilde{\Phi} = \{\tilde{\varphi}_0, \tilde{\varphi}_1, \tilde{\varphi}_2\}$  is derived in Example 1.41.

together with  $\Phi$  forms a biorthogonal pair of bases for  $H$ . The synthesis operator for this basis is given by

$$\tilde{\Phi} = \Phi A = \Phi(\Phi^* \Phi)^{-1}. \quad (1.122b)$$

Recall from (1.116) that the synthesis operator associated with the dual basis could be written as  $\tilde{\Phi} = (\Phi^{-1})^*$ . However, the inverse and adjoint in this expression are difficult to interpret. In contrast, the key virtue of (1.122) is that the inversion is of the Gram matrix, which is easier to interpret because it is an operator from  $\ell^2(\mathcal{K})$  to  $\ell^2(\mathcal{K})$ . This is illustrated in the following finite-dimensional example.

**EXAMPLE 1.41 (DUAL OF BASIS OF PERIODIC HAT FUNCTIONS)** Consider the function

$$\varphi_0(t) = \begin{cases} t, & \text{for } t \in [0, 1]; \\ 2 - t, & \text{for } t \in (1, 2]; \\ 0, & \text{for } t \in (2, 3] \end{cases} \quad (1.123)$$

in  $\mathcal{L}^2([0, 3])$  and its circular shifts by 1, as shown in Figure 1.24. The set  $\Phi = \{\varphi_0, \varphi_1, \varphi_2\}$  is a basis for a subspace  $S = \overline{\text{span}}(\Phi) \subset \mathcal{L}^2([0, 3])$ . This subspace is the set of functions  $x$  that satisfy  $x(0) = x(3)$  and are piecewise linear on  $[0, 3]$  with breakpoints at 1 and 2.

We wish to find the basis  $\tilde{\Phi} = \{\tilde{\varphi}_0, \tilde{\varphi}_1, \tilde{\varphi}_2\}$  that forms a biorthogonal pair with  $\Phi$ . The Gram matrix of  $\Phi$  is

$$G = \begin{bmatrix} 2/3 & 1/6 & 1/6 \\ 1/6 & 2/3 & 1/6 \\ 1/6 & 1/6 & 2/3 \end{bmatrix}. \quad (1.124)$$

Using its inverse in (1.122a) yields

$$\begin{aligned}\tilde{\varphi}_0 &= \frac{5}{3}\varphi_0 - \frac{1}{3}\varphi_1 - \frac{1}{3}\varphi_2, \\ \tilde{\varphi}_1 &= -\frac{1}{3}\varphi_0 + \frac{5}{3}\varphi_1 - \frac{1}{3}\varphi_2, \\ \tilde{\varphi}_2 &= -\frac{1}{3}\varphi_0 - \frac{1}{3}\varphi_1 + \frac{5}{3}\varphi_2.\end{aligned}$$

These functions are depicted in Figure 1.24. Since each  $\tilde{\varphi}_k$  is a linear combination of  $\varphi_k$ s, it is clear that  $\text{span}(\tilde{\Phi}) \subseteq \text{span}(\Phi)$ . One can also show that  $\text{span}(\Phi) \subseteq \text{span}(\tilde{\Phi})$ . For an intuitive understanding, note that each  $\tilde{\varphi}_k$  satisfies  $\tilde{\varphi}_k(0) = \tilde{\varphi}_k(3)$  and is piecewise linear on  $[0, 3]$  with breakpoints at 1 and 2; thus, the sets span the same subspace. The solution is unique, whereas many sets of functions satisfy the biorthogonality condition (1.102) without satisfying  $\text{span}(\tilde{\Phi}) = \text{span}(\Phi)$ .

The dual of the dual of a basis is the original basis, and a basis is its own dual if and only if it is an orthonormal basis. Also, if the Riesz basis constants of the basis  $\Phi$  are  $\lambda_{\min}$  and  $\lambda_{\max}$ , then  $\tilde{\Phi}$  is a Riesz basis with constants  $1/\lambda_{\max}$  and  $1/\lambda_{\min}$ . Establishing these facts formally is left for Exercise 1.44. As we mentioned earlier, it can be advantageous for numerical computations to have  $\lambda_{\min} \approx \lambda_{\max}$ . The property of  $\lambda_{\min}/\lambda_{\max} \approx 1$  is then maintained by the dual.

**Dual Coefficients** As we noted earlier in reference to computation of inner products, it is often convenient to use only one basis explicitly. Then, we face the problem of finding expansions with respect to the basis  $\Phi$  from analysis with  $\Phi^*$ . Unless  $\Phi^* = \tilde{\Phi}^*$ , in which case  $\Phi$  is an orthonormal basis, the coefficients obtained with analysis by  $\Phi^*$  must be adjusted to be the right ones to use in synthesis by  $\Phi$ . The adjustment of coefficients is analogous to computation of the dual basis.

To have an expansion with respect to the basis  $\Phi$  from analysis with  $\Phi^*$ , we seek an operator  $A : \ell^2(\mathcal{K}) \rightarrow \ell^2(\mathcal{K})$  such that

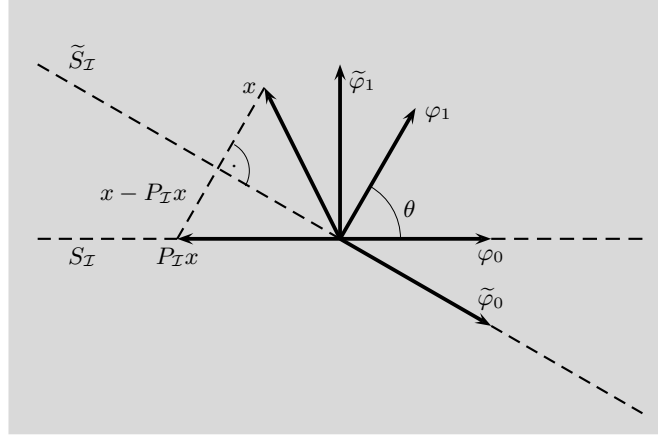
$$x = \Phi A \Phi^* x \quad \text{for every } x \in H.$$

It is easy to verify that  $A = (\Phi^* \Phi)^{-1}$ , the inverse of the Gram matrix, is the desired operator. In terms of the coefficient sequences defined in (1.104b) and (1.106),  $A$  maps  $\tilde{\alpha}$  to  $\alpha$ . Naturally, the Gram matrix maps  $\alpha$  to  $\tilde{\alpha}$ .

**Oblique Projection** Similarly to the truncation of an orthonormal expansion giving an orthogonal projection (Theorem 1.40), truncation of (1.105a) or (1.106) gives an oblique projection. The proof of the following result is left to Exercise 1.45.

**THEOREM 1.46** Given sets  $\Phi_{\mathcal{I}} = \{\varphi_k\}_{k \in \mathcal{I}} \subset H$  and  $\tilde{\Phi}_{\mathcal{I}} = \{\tilde{\varphi}_k\}_{k \in \mathcal{I}} \subset H$  satisfying

$$\langle \varphi_i, \tilde{\varphi}_k \rangle = \delta_{i-k} \quad \text{for every } i, k \in \mathcal{I},$$



**Figure 1.25:** Example of an oblique projection. The projection is onto  $S_{\mathcal{I}}$ , the subspace spanned by  $\varphi_0$ . The projection is orthogonal to  $\tilde{S}_{\mathcal{I}}$ , the subspace spanned by the biorthogonal vector  $\tilde{\varphi}_0$ .

for any  $x$  in  $H$ ,

$$P_{\mathcal{I}} x = \sum_{k \in \mathcal{I}} \langle x, \tilde{\varphi}_k \rangle \varphi_k \quad (1.125a)$$

$$= \Phi_{\mathcal{I}} \tilde{\Phi}_{\mathcal{I}}^* x \quad (1.125b)$$

is a projection of  $x$  onto  $S_{\mathcal{I}} = \overline{\text{span}}(\{\varphi_k\}_{k \in \mathcal{I}})$ . The residual satisfies  $x - P_{\mathcal{I}} x \perp \tilde{S}_{\mathcal{I}}$ , where  $\tilde{S}_{\mathcal{I}} = \overline{\text{span}}(\{\tilde{\varphi}_k\}_{k \in \mathcal{I}})$ .

EXAMPLE 1.42 (OBLIQUE PROJECTION, EXAMPLE 1.40 CONT'D) We continue our discussion of Example 1.40 and illustrate oblique projection. Define  $P_{\mathcal{I}}$  via (1.125) as

$$P_{\mathcal{I}} x = \langle x, \tilde{\varphi}_0 \rangle \varphi_0 = \Phi_{\mathcal{I}} \tilde{\Phi}_{\mathcal{I}}^* x,$$

with

$$\Phi_{\mathcal{I}} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T, \quad \tilde{\Phi}_{\mathcal{I}} = \begin{bmatrix} 1 & -\cot \theta \end{bmatrix}^T.$$

Figure 1.25 illustrates the projection (not orthogonal anymore), the subspace  $S_{\mathcal{I}}$ , the residual  $x - P_{\mathcal{I}} x$ , and the subspace  $\tilde{S}_{\mathcal{I}}$ .

While the above theorem gives an important property, it is not as useful as Theorem 1.40 because oblique projections do not solve best approximation problems.

**Decomposition** By applying Theorem 1.46 with any single-element set  $\mathcal{I}$ , we see that any one term of (1.105a) or (1.106) is an oblique projection onto a 1-

dimensional subspace. Thus, a biorthogonal pair of bases induces a pair of decompositions

$$H = \bigoplus_{k \in \mathcal{K}} S_{\{k\}} \quad \text{and} \quad H = \bigoplus_{k \in \mathcal{K}} \tilde{S}_{\{k\}}$$

where  $S_{\{k\}} = \text{span}(\varphi_k)$  and  $\tilde{S}_{\{k\}} = \text{span}(\tilde{\varphi}_k)$ . Actually, because of linear independence and completeness, any basis gives a decomposition of the form above. A key merit of a decomposition that comes from a biorthogonal pair of bases is that the expansion coefficients are determined simply as in (1.104).

**Best Approximation and the Normal Equations** According to the projection theorem (Theorem 1.26), given a closed subspace  $S$  of a Hilbert space  $H$ , the best approximation of a vector  $x$  in  $H$  is given by the orthogonal projection of  $x$  onto  $S$ . In Theorem 1.40, we saw how to compute this orthogonal projection in one special case. We now derive a general methodology using bases.

Denote the orthogonal projection of  $x$  onto  $S$  by  $\hat{x}$ . According to the projection theorem,  $\hat{x}$  is uniquely determined by  $\hat{x} \in S$  and  $x - \hat{x} \perp S$ . Given a basis  $\{\varphi_k\}_{k \in \mathcal{I}}$  for  $S$ , the projection being in  $S$  is ensured by

$$\hat{x} = \sum_{k \in \mathcal{I}} \beta_k \varphi_k \quad (1.126a)$$

for some coefficient sequence  $\beta$ , and the residual being orthogonal to  $S$  is expressed as

$$\langle x - \hat{x}, \varphi_i \rangle = 0 \quad \text{for every } i \in \mathcal{I}. \quad (1.126b)$$

Substituting (1.126a) into (1.126b) gives

$$\langle x, \varphi_i \rangle = \langle \sum_{k \in \mathcal{I}} \beta_k \varphi_k, \varphi_i \rangle = \sum_{k \in \mathcal{I}} \beta_k \langle \varphi_k, \varphi_i \rangle \quad \text{for every } i \in \mathcal{I}.$$

Solving these equations gives the following result:

**THEOREM 1.47 (NORMAL EQUATIONS)** Given a vector  $x$  and a linearly independent set  $\{\varphi_k\}_{k \in \mathcal{I}}$  in a separable Hilbert space  $H$ , the vector closest to  $x$  in  $\overline{\text{span}}(\{\varphi_k\}_{k \in \mathcal{I}})$  is

$$\hat{x} = \sum_{k \in \mathcal{I}} \beta_k \varphi_k \quad (1.127a)$$

$$= \Phi \beta, \quad (1.127b)$$

where  $\beta$  is the unique solution to the system of equations

$$\sum_{k \in \mathcal{I}} \beta_k \langle \varphi_k, \varphi_i \rangle = \langle x, \varphi_i \rangle \quad \text{for every } i \in \mathcal{I}, \quad \text{or,} \quad (1.128a)$$

$$\Phi^* \Phi \beta = \Phi^* x. \quad (1.128b)$$

## 1.5. Bases and Frames

93

Equations (1.128) are called *normal equations* because they express the normality (orthogonality) of the residual and the subspace (from (1.126b)). In operator notation, combining (1.127b) and (1.128b) leads to

$$\hat{x} = \Phi(\Phi^*\Phi)^{-1}\Phi^*x = Px. \quad (1.129)$$

It is then easy to check that  $P$  is an orthogonal projection operator (Exercise 1.46). Invertibility of the Gram matrix  $\Phi^*\Phi$  follows from the linear independence of  $\{\varphi_k\}_{k \in \mathcal{I}}$  (Exercise 1.46). If the set  $\{\varphi_k\}_{k \in \mathcal{I}}$  is not linearly independent, the projection theorem still ensures that  $\hat{x}$  is unique, but naturally its expansion with respect to  $\{\varphi_k\}_{k \in \mathcal{I}}$  is not unique. We illustrate these concepts with an example.

EXAMPLE 1.43 (NORMAL EQUATIONS IN  $\mathbb{R}^3$ ) Let  $\varphi_0 = [1, 1, 0]^T$  and  $\varphi_1 = [0, 1, 1]^T$ . Given a vector  $x = [1, 1, 1]^T$ , according to Theorem 1.47, the vector in  $\text{span}(\{\varphi_0, \varphi_1\})$  closest to  $x$  is

$$\hat{x} = \beta_0\varphi_0 + \beta_1\varphi_1,$$

with  $\beta$  the unique solution to (1.128b), which simplifies to

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \end{bmatrix}.$$

Solving the system yields  $\beta_0 = \beta_1 = 2/3$ , leading to

$$\hat{x} = \frac{2}{3}(\varphi_0 + \varphi_1) = \begin{bmatrix} 2/3 \\ 4/3 \\ 2/3 \end{bmatrix}.$$

We can easily check that the residual  $x - \hat{x}$  is orthogonal to  $\text{span}(\{\varphi_0, \varphi_1\})$ :

$$x - \hat{x} = \begin{bmatrix} 2/3 \\ -2/3 \\ 2/3 \end{bmatrix} \perp \alpha_0\varphi_0 + \alpha_1\varphi_1 = \begin{bmatrix} \alpha_0 \\ \alpha_0 + \alpha_1 \\ \alpha_1 \end{bmatrix}.$$

Now let  $\varphi_2 = [1, 0, -1]^T$ . The vector in  $\text{span}(\{\varphi_0, \varphi_1, \varphi_2\})$  closest to  $x$  is

$$\hat{x} = \beta_0\varphi_0 + \beta_1\varphi_1 + \beta_2\varphi_2$$

where  $\beta$  satisfies

$$\begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & -1 \\ 1 & -1 & 2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 0 \end{bmatrix}.$$

The solutions for  $\beta$  are not unique, but all solutions yield  $\hat{x} = [2/3, 4/3, 2/3]^T$  as before. This is as expected since  $\text{span}(\{\varphi_0, \varphi_1, \varphi_2\}) = \text{span}(\{\varphi_0, \varphi_1\})$ .

In the special case that  $\{\varphi_k\}_{k \in \mathcal{I}}$  is an orthonormal set, the normal equations (1.128) simplify greatly:

$$\langle x, \varphi_i \rangle \stackrel{(a)}{=} \sum_{k \in \mathcal{I}} \beta_k \langle \varphi_k, \varphi_i \rangle \stackrel{(b)}{=} \sum_{k \in \mathcal{I}} \beta_k \delta_{k-i} \stackrel{(c)}{=} \beta_i, \quad \text{for every } i \in \mathcal{I},$$

where (a) is (1.128); (b) follows from orthonormality; and (c) from the definition of the Kronecker delta sequence, (1.9). Thus the coefficients of the expansion of  $\hat{x}$  with respect to  $\{\varphi_k\}_{k \in \mathcal{I}}$  come from analysis with the same set of vectors, exactly as in Theorem 1.40.

Solving the normal equations is also simplified by having available  $\{\tilde{\varphi}_k\}_{k \in \mathcal{I}}$  that together with  $\{\varphi_k\}_{k \in \mathcal{I}}$  forms a biorthogonal pair of bases for the subspace of interest. Then by combination of (1.122b) and (1.129), the best approximation is  $\hat{x} = \tilde{\Phi} \Phi^* x$ .

In general,  $\{\varphi_k\}_{k \in \mathcal{I}}$  is not an orthonormal set, and we may want to express both  $x$  and  $\hat{x}$  through expansions with a different basis  $\{\psi_k\}_{k \in \mathcal{K}}$  that is not necessarily orthonormal:

$$x = \sum_{k \in \mathcal{K}} \alpha_k \psi_k \quad \text{and} \quad \hat{x} = \sum_{k \in \mathcal{K}} \hat{\alpha}_k \psi_k.$$

Properties of the mapping from  $\alpha$  to  $\hat{\alpha}$  are established in Exercise 1.47. In particular, orthogonal projection in  $H$  corresponds to orthogonal projection in the coefficient space if and only if expansions are with respect to an orthonormal basis.

**Successive Approximation** Continuing our discussion of best approximation, now consider the computation of a sequence of best approximations in subspaces of increasing dimension. Let  $\{\varphi_i\}_{i \in \mathbb{N}}$  be a linearly independent set, and for each  $k \in \mathbb{N}$ , let  $S_k = \text{span}(\{\varphi_0, \varphi_1, \dots, \varphi_{k-1}\})$ . Let  $\hat{x}^{(k)}$  denote the best approximation of  $x$  in  $S_k$ .<sup>21</sup> In Section 1.5.2, we found a simple recursive computation for the expansion of  $\hat{x}^{(k)}$  with respect to  $\{\varphi_i\}_{i \in \mathbb{N}}$  for the case that  $\{\varphi_i\}_{i \in \mathbb{N}}$  is an orthonormal set; see (1.99). Here, recursive computation is made more complicated by the lack of orthonormality of the basis.

The spaces  $S_k$  and  $S_{k+1}$  are nested with  $S_k \subset S_{k+1}$ , so approximating  $x$  with a vector from  $S_{k+1}$  instead of one from  $S_k$  cannot make the approximation quality worse; improvement is obtained by capturing the component of  $x$  that could not be captured before. The nesting of subspaces can be expressed as

$$S_{k+1} = S_k \oplus T_k, \tag{1.130}$$

where the one-dimensional subspace  $T_k$  is not uniquely specified. If we choose  $T_k$  to make this direct sum an orthogonal decomposition, then the increment  $\hat{x}^{(k+1)} - \hat{x}^{(k)}$  will simply be the orthogonal projection of  $x$  onto  $T_k$ . The decomposition (1.130) is orthogonal when  $T_k = \text{span}(\psi_k)$  with  $\psi_k \perp S_k$ , so we get the desired direct sum by choosing  $\psi_k$  parallel to the residual in orthogonally projecting  $\varphi_k$  to  $S_k$ . This approach simplifies the computation of the increment at the cost of requiring  $\psi_k$ . It can yield a computation savings when the entire sequence of approximations is desired and the  $\psi_k$ s are computed recursively through Gram–Schmidt orthogonalization.

Let  $\hat{x}^{(0)} = \mathbf{0}$ , and for  $k = 0, 1, \dots$ , make the following computations. First, compute  $\psi_k$  orthogonal to  $S_k$  and, for convenience in other computations, of unit

<sup>21</sup>By these definitions,  $S_0 = \{\mathbf{0}\}$  and  $\hat{x}^{(0)} = \mathbf{0}$ .

norm:

$$v_k = \sum_{i=0}^{k-1} \langle \varphi_k, \psi_i \rangle \psi_i, \quad (1.131a)$$

$$\psi_k = \frac{\varphi_k - v_k}{\|\varphi_k - v_k\|}. \quad (1.131b)$$

In this computation,  $v_k$  is the orthogonal projection of  $\varphi_k$  onto  $S_k$  since  $\{\psi_i\}_{i=0}^{k-1}$  is an orthonormal basis for  $S_k$ ; see (1.97a). With this intermediate orthogonalization, we have

$$\widehat{x}^{(k+1)} = \widehat{x}^{(k)} + \langle x, \psi_k \rangle \psi_k. \quad (1.131c)$$

Exercise 1.48 explores the connection between this algorithm and the normal equations.

### 1.5.4 Frames

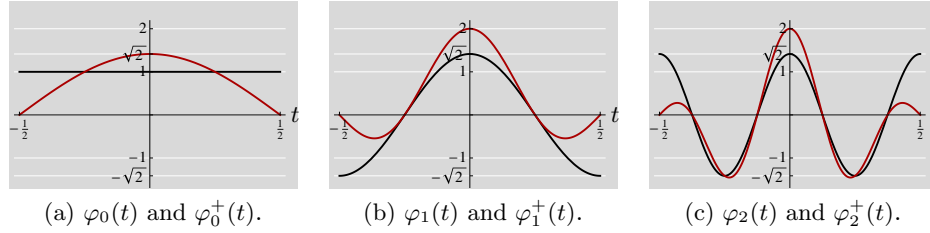
Bases are sets of vectors that are both linearly independent and complete (see Definition 1.33). Completeness ensures the existence of expansions; linear independence ensures that the expansions are unique. In a finite-dimensional space, linear independence upper bounds the number of vectors by the dimension of the space while completeness lower bounds the number of vectors by the dimension of the space; thus, there are exactly as many vectors as the dimension of the space. Frames are more general than bases because they are complete but not necessarily linearly independent. In a finite-dimensional space, a frame must have at least as many vectors as the dimension of the space. In infinite-dimensional spaces, a frame must have infinitely many vectors, and imposing something analogous to the Riesz basis condition (1.80) prevents certain pathologies.

Why would we want more than the minimum number of vectors for completeness? There are several possible disadvantages: When linear independence is lost, uniqueness of expansions is lost with it, and it would seem at first glance that having a larger set of vectors implies more computations in both analysis and synthesis. The primary advantages come from flexibility in design: Fixing analysis leaves flexibility in synthesis and vice versa; and we will see in Chapters 10, 11, and 12 that frames can have additional desirable properties unavailable with bases.

**DEFINITION 1.48 (FRAME)** The set of vectors  $\Phi = \{\varphi_k\}_{k \in \mathcal{J}} \subset H$ , where  $\mathcal{J}$  is finite or countably infinite, is called a frame for Hilbert space  $H$  when the largest  $\lambda_{\min}$  and smallest  $\lambda_{\max}$  such that

$$\lambda_{\min} \|x\|^2 \leq \sum_{k \in \mathcal{J}} |\langle x, \varphi_k \rangle|^2 \leq \lambda_{\max} \|x\|^2, \quad \text{for every } x \text{ in } H, \quad (1.132)$$

satisfy  $0 < \lambda_{\min} \leq \lambda_{\max} < \infty$ . The constants  $\lambda_{\min}$  and  $\lambda_{\max}$  are called frame bounds.



**Figure 1.26:** Example frame functions from  $\Phi \cup \Phi^+$  (black for functions from  $\Phi$  and red for functions from  $\Phi^+$ ).

A frame is sometimes called a *Riesz sequence*. This highlights the similarity of (1.132) to condition (1.80) in Definition 1.34 for Riesz bases and also that a frame is not necessarily a basis.

Let us immediately compare and contrast Definitions 1.34 and 1.48:

- (i) The notation for the index set has been changed from  $\mathcal{K}$  to  $\mathcal{J}$  to reflect that these are not generally the same size when  $H$  has finite dimension.
- (ii) The definition of a frame in Definition 1.48 uses the set  $\Phi$  in analysis; in contrast, the definition of a basis in Definition 1.34 uses the set  $\Phi$  in synthesis. Nevertheless, both bases and frames can be used for both analysis and synthesis. A frame generally lacks linear independence, so an expansion of the form  $x = \sum_{k \in \mathcal{J}} \alpha_k \varphi_k$  is not necessarily unique; this prevents a closer parallel in the definitions.
- (iii) If  $\Phi$  and  $\tilde{\Phi}$  form a biorthogonal pair of bases, then the unique expansion with respect to  $\tilde{\Phi}$  is obtained through analysis with  $\Phi$ ; see (1.107). In this case, comparison of Definitions 1.34 and 1.48 shows that  $\tilde{\Phi}$  being a Riesz basis with constants  $\lambda_{\min}$  and  $\lambda_{\max}$  implies that  $\Phi$  is a frame with frame bounds  $\lambda_{\min}$  and  $\lambda_{\max}$ . Since the dual of a Riesz basis is a Riesz basis, we reach the simple conclusion that any Riesz basis is a frame.

Uses of frames in analysis and synthesis will be established shortly. Exercise 1.49 explores the differences between Definitions 1.34 and 1.48 further.

**EXAMPLE 1.44 (FRAME OF COSINE FUNCTIONS)** Starting with  $\Phi = \{\varphi_k\}_{k \in \mathbb{N}} \subset \mathcal{L}^2([-1/2, 1/2])$  from (1.21), define the set of functions  $\Phi^+ = \{\varphi_k^+\}_{k \in \mathbb{N}}$  by multiplying each  $\varphi_k$  by  $\sqrt{2} \cos(\pi t)$ :

$$\varphi_k^+(t) = \sqrt{2} \cos(\pi t) \varphi_k(t), \quad k \in \mathbb{N}.$$

A few functions from  $\Phi \cup \Phi^+$  are shown in Figure 1.26.

We know already from Example 1.31 that  $\Phi$  is an orthonormal basis for the closure of its span,  $S = \overline{\text{span}}(\Phi)$ . The union  $\Phi \cup \Phi^+$  is a frame for  $S$ . To see that the closure of the span of  $\Phi \cup \Phi^+$  is not larger than  $S$ , note that each  $\varphi_k^+$  can be



## 1.5. Bases and Frames

97

written as a linear combination of elements of  $\Phi$ . For  $k \in \mathbb{Z}^+$ ,

$$\begin{aligned}\varphi_k^+(t) &= \sqrt{2} \cos(\pi t) \varphi_k(t) = 2 \cos(\pi t) \cos(2\pi kt) \\ &= \cos(2\pi(k-1)t) + \cos(2\pi(k+1)t) \\ &= \begin{cases} \varphi_{k-1}(t) + \frac{1}{\sqrt{2}} \varphi_{k+1}(t), & \text{for } k = 1; \\ \frac{1}{\sqrt{2}} \varphi_{k-1}(t) + \frac{1}{\sqrt{2}} \varphi_{k+1}(t), & \text{for } k = 2, 3, \dots \end{cases}\end{aligned}$$

Being able to write  $\varphi_0^+$  as a linear combination of  $\{\varphi_k\}_{k \in \mathbb{N}}$  becomes clear from the Fourier series studied in Section 3.5. Computing the frame bounds of this frame is left for Exercise 1.50.

**Operators Associated with Frames** Analogously to bases, we can define the *synthesis operator* associated with  $\{\varphi_k\}_{k \in \mathcal{J}}$  to be

$$\Phi : \ell^2(\mathcal{J}) \rightarrow H, \quad \text{with} \quad \Phi \alpha = \sum_{k \in \mathcal{J}} \alpha_k \varphi_k. \quad (1.133)$$

The second inequality of (1.132) implies that the norm of this linear operator is finite and the operator thus bounded.

Similarly, we define the *analysis operator* associated with  $\{\varphi_k\}_{k \in \mathcal{J}}$  to be

$$\Phi^* : H \rightarrow \ell^2(\mathcal{J}), \quad \text{with} \quad (\Phi^* x)_k = \langle x, \varphi_k \rangle, \quad k \in \mathcal{J}. \quad (1.134)$$

The norm of the analysis operator is the same as that of the synthesis operator.

The power of the operator notation can be seen in rephrasing (1.132) as

$$\lambda_{\min} I \leq \Phi \Phi^* \leq \lambda_{\max} I. \quad (1.135)$$

The first equality can be derived as follows:

$$\begin{aligned}\langle (\Phi \Phi^* - \lambda_{\min} I)x, x \rangle &\stackrel{(a)}{=} \langle \Phi \Phi^* x, x \rangle - \langle \lambda_{\min} I x, x \rangle \stackrel{(b)}{=} \langle \Phi \Phi^* x, x \rangle - \lambda_{\min} \langle x, x \rangle \\ &\stackrel{(c)}{=} \langle \Phi^* x, \Phi^* x \rangle - \lambda_{\min} \langle x, x \rangle = \|\Phi^* x\|^2 - \lambda_{\min} \|x\|^2 \stackrel{(d)}{\geq} 0,\end{aligned}$$

where (a) follows from distributivity of the inner product; (b) from the linearity in the first argument of the inner product and the meaning of the identity operator; (c) from the definition of adjoint; and (d) from the first inequality of (1.132). The second inequality of (1.135) can be derived similarly.

Because  $\Phi \Phi^*$  is a Hermitian operator, the operator analogue of (1.225) holds; thus, the frame bounds are the smallest and largest eigenvalues of  $\Phi \Phi^*$ . This gives an easy way to find the frame bounds, as we illustrate in the following example.

**EXAMPLE 1.45 (FRAMES IN  $\mathbb{R}^2$ )** In (1.14), we defined a frame. The vectors are clearly not linearly independent; however, they do satisfy (1.132). To compute the frame bounds, we could follow the path from Example 1.28: compute  $\sum_{k \in \mathcal{J}} |\langle x, \varphi_k \rangle|^2$  and find the frame bounds as infimum and supremum of  $\sum_{k \in \mathcal{J}} |\langle x, \varphi_k \rangle|^2 / (x_0^2 + x_1^2)$ . A much easier way is to use the operator notation,

where  $\Phi$  is given in (1.16a). This  $\Phi$  is a rectangular matrix, illustrating the fact that a frame is an overcomplete expansion. Then  $\Phi\Phi^*$  is

$$\Phi\Phi^* = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} = \underbrace{\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}}_V \underbrace{\begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}}_\Lambda \underbrace{\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}}_{V^{-1}},$$

where we have performed an eigendecomposition on the Hermitian matrix  $\Phi\Phi^*$  via (1.210a). We can immediately read the smallest and largest eigenvalues,  $\lambda_{\min} = 1$  and  $\lambda_{\max} = 3$ , as the frame bounds.

In many ways, including expansion and inner product computation, frames play the same roles as bases. When a frame lacks linear independence, it cannot induce a subspace decomposition because of the uniqueness requirement in Definition 1.30. The connection between frames and projections is more subtle. We now develop these ideas further, covering the special case of tight frames first.

### Tight Frames

**DEFINITION 1.49 (TIGHT FRAME)** The frame  $\Phi = \{\varphi_k\}_{k \in \mathcal{J}} \subset H$ , where  $\mathcal{J}$  is finite or countably infinite, is called a tight frame, or a  $\lambda$ -tight frame, for Hilbert space  $H$  when its frame bounds are equal,  $\lambda_{\min} = \lambda_{\max} = \lambda$ .

For a  $\lambda$ -tight frame, (1.135) simplifies to

$$\Phi\Phi^* = \lambda I. \quad (1.136)$$

A tight frame is a counterpart of an orthonormal basis, as we will see shortly.

**EXAMPLE 1.46 (FINITE-DIMENSIONAL TIGHT FRAME)** Take the following three vectors as a frame for  $\mathbb{R}^2$ :

$$\varphi_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \varphi_1 = \begin{bmatrix} -\frac{1}{2} \\ \frac{\sqrt{3}}{2} \end{bmatrix}, \quad \varphi_2 = \begin{bmatrix} -\frac{1}{2} \\ -\frac{\sqrt{3}}{2} \end{bmatrix}, \quad (1.137a)$$

$$\Phi = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}. \quad (1.137b)$$

Computing its frame bounds as we did in Example 1.45, we find that

$$\Phi\Phi^* = \frac{3}{2}I,$$

and thus, the eigenvalues are  $\lambda_{\min} = \lambda_{\max} = 3/2$ , and the frame is tight. Note that this frame is just a normalized version of the one in (1.15), which is a 1-tight frame.

Frames that are 1-tight are called *Parseval tight frames*. We can normalize any  $\lambda$ -tight frame by pulling  $1/\sqrt{\lambda}$  into the sum in (1.132) to yield a 1-tight frame,

$$\sum_k \left| \langle x, \lambda^{-1/2} \tilde{\varphi}_k \rangle \right|^2 = \sum_k |\langle x, \tilde{\varphi}'_k \rangle|^2 = \|x\|^2. \quad (1.138)$$

Because of this normalization, we can associate a 1-tight frame to any tight frame. Note that orthonormal bases are 1-tight frames with all unit-norm vectors. In general, the vectors in a 1-tight frame do not have unit norms or even equal norms.

**Expansion and Inner Product Computation** Expansion coefficients with respect to a 1-tight frame can be obtained by using the same 1-tight frame for signal analysis.

**THEOREM 1.50 (TIGHT FRAME EXPANSIONS)** Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{J}}$  be a 1-tight frame for Hilbert space  $H$ . Analysis of any  $x$  in  $H$  gives expansion coefficients in  $\ell^2(\mathcal{J})$

$$\alpha_k = \langle x, \varphi_k \rangle \quad \text{for } k \in \mathcal{J}, \quad \text{or,} \quad (1.139a)$$

$$\alpha = \Phi^* x. \quad (1.139b)$$

Synthesis with these coefficients yields

$$x = \sum_{k \in \mathcal{J}} \langle x, \varphi_k \rangle \varphi_k \quad (1.140a)$$

$$= \Phi \alpha = \Phi \Phi^* x. \quad (1.140b)$$

Note the apparent similarity of this theorem to Theorem 1.38. The equations in these theorems are identical, and each theorem shows that analysis and synthesis with the same set of vectors yields an identity on  $H$ . In the orthonormal basis case, the expansion is *unique*; in the 1-tight frame case it is generally not. The  $\alpha$  given by (1.139b) can be replaced by any  $\alpha' = \alpha + \alpha^\perp$ , where  $\alpha^\perp$  is in the null space of  $\Phi$ , while maintaining  $x = \Phi \alpha'$ .

The theorem follows from two simple facts:  $\alpha \in \ell^2(\mathcal{J})$  because  $\Phi^*$  is a bounded operator; and

$$\Phi \Phi^* = I \quad \text{on } H \quad (1.141)$$

by setting  $\lambda = 1$  in (1.136). This leads to Parseval's equalities for tight frames:

**THEOREM 1.51 (PARSEVAL'S EQUALITIES FOR 1-TIGHT FRAMES)** Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{J}}$  be a 1-tight frame for Hilbert space  $H$ . Expansion with coefficients (1.139) satisfies

$$\|x\|^2 = \sum_{k \in \mathcal{J}} |\langle x, \varphi_k \rangle|^2 \quad (1.142a)$$

$$= \|\Phi^* x\|^2 = \|\alpha\|^2. \quad (1.142b)$$

More generally,

$$\langle x, y \rangle = \sum_{k \in \mathcal{J}} \langle x, \varphi_k \rangle \langle y, \varphi_k \rangle^* \quad (1.143a)$$

$$= \langle \Phi^* x, \Phi^* y \rangle = \langle \alpha, \beta \rangle. \quad (1.143b)$$

Again, this theorem looks formally the same as Theorem 1.39. Beyond what we have already mentioned ( $\Phi$  is a linearly-dependent set of vectors and the expansion is nonunique), this theorem hides even more. For example, the norm-preservation property could be misleading; the frame in the theorem is 1-tight, and thus, even if its elements have all the same norm, that norm is generally not 1.

**EXAMPLE 1.47 (PARSEVAL'S EQUALITIES FOR TIGHT FRAMES)** Let us continue with the frame from (1.15). Its vectors are all of norm  $2/3$ . Normalizing it so that all of its vectors are of unit norm yields the frame in (1.137). Computing the norm of the expansion coefficients  $\|\alpha\|^2$  for this frame yields

$$\|\alpha\|^2 = \|\Phi^* x\|^2 = \left\| \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} \right\|^2 = \left\| \begin{bmatrix} x_0 \\ -\frac{x_0 - \sqrt{3}x_1}{2} \\ -\frac{x_0 + \sqrt{3}x_1}{2} \end{bmatrix} \right\|^2 = \frac{3}{2} \|x\|^2.$$

This tells us that for this tight frame with all unit-norm vectors, the norm of the expansion coefficients is  $3/2$  times larger than that of the vector itself. This is intuitive as we have  $3/2$  times more vectors than needed for an expansion in  $\mathbb{R}^2$ .

This example generalizes to all finite-dimensional tight frames with unit-norm vectors. For such frames, the factor appearing in the Parseval's equality denotes the *redundancy* of the frame.

**Inverse Synthesis and Analysis** For a 1-tight frame, (1.141) shows that the synthesis operator is a left inverse of the analysis operator. Unlike with an orthonormal basis, the synthesis operator associated with a 1-tight frame is generally not a right inverse (hence, not an inverse) of the analysis operator because  $\Phi^* \Phi \neq I$ . In finite dimensions, this can be seen easily from the rank of  $\Phi^* \Phi$ ; the rank of  $\Phi^* \Phi$  is the dimension of  $H$ , but  $\Phi^* \Phi$  is an operator on  $\mathbb{C}^{|\mathcal{J}|}$  where  $|\mathcal{J}|$  may be larger than the dimension of  $H$ .

**EXAMPLE 1.48 (INVERSE RELATIONSHIP FOR FRAME OPERATORS)** Continue with the 1-tight frame from (1.15). We have already seen that  $\Phi \Phi^* = I_{2 \times 2}$ . We also have

$$\Phi^* \Phi = \begin{bmatrix} 2/3 & -1/3 & -1/3 \\ -1/3 & 2/3 & -1/3 \\ -1/3 & -1/3 & 2/3 \end{bmatrix} \neq I_{3 \times 3}.$$

The rank of  $\Phi^* \Phi$  is 2.

**Orthogonal Projection** Since a frame for  $H$  generally has more than the minimum number of vectors needed to span  $H$ , omitting some terms from the synthesis sum (1.140) does not necessarily restrict the result to a proper subspace of  $H$ . Thus, a frame (even a 1-tight frame) does not yield a result analogous to Theorem 1.40 for computing orthogonal projections on  $H$ .

A different orthogonal projection property is easy to verify for any 1-tight frame:  $\Phi^* \Phi : \ell^2(\mathcal{J}) \rightarrow \ell^2(\mathcal{J})$  is the orthogonal projection onto  $\mathcal{R}(\Phi^*)$ . This has important consequences for robustness to noise that are developed fully in later chapters.

### General Frames

Tight frames are a small class of frames as defined in Definition 1.48. A frame may generally have frame bounds that differ, the distance between which gives us information about the quality of the frame.

**Dual Frame Pairs and Expansion** When a frame  $\Phi$  is not 1-tight, to find expansions coefficients with respect to  $\Phi$  with a linear operator requires a second frame, in analogy to biorthogonal pairs of bases.

**DEFINITION 1.52 (DUAL PAIR OF FRAMES)** The sets of vectors  $\Phi = \{\varphi_k\}_{k \in \mathcal{J}} \subset H$  and  $\tilde{\Phi} = \{\tilde{\varphi}_k\}_{k \in \mathcal{J}} \subset H$ , where  $\mathcal{J}$  is finite or countably infinite, are called a dual pair of frames for Hilbert space  $H$  when

- (i) each is a *frame* for  $H$ ; and
- (ii) for any  $x$  in  $H$ ,

$$x = \sum_{k \in \mathcal{K}} \langle x, \tilde{\varphi}_k \rangle \varphi_k \quad (1.144a)$$

$$= \Phi \tilde{\Phi}^* x. \quad (1.144b)$$

Note that this definition combines the roles of Definition 1.42 and Theorem 1.43 for general frames. This is necessary because no simple pairwise condition between vectors like (1.102) will imply (1.144).

**EXAMPLE 1.49 (DUAL PAIRS OF FRAMES IN  $\mathbb{R}^2$ )** Let  $\Phi$  be the frame for  $\mathbb{R}^2$  defined in (1.14). Since synthesis operator  $\Phi$  is a  $2 \times 3$  matrix with rank 2, it has infinitely many right inverses; any right inverse specifies a frame that forms a dual pair with  $\Phi$ . Examples include the following:

$$\left\{ \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 2 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right\}, \quad \left\{ \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ 2 \end{bmatrix}, \begin{bmatrix} -1 \\ 1 \end{bmatrix} \right\}, \quad \left\{ \begin{bmatrix} 2/3 \\ -1/3 \end{bmatrix}, \begin{bmatrix} -1/3 \\ 2/3 \end{bmatrix}, \begin{bmatrix} -1/3 \\ -1/3 \end{bmatrix} \right\}.$$

The first of these examples demonstrates that a frame can have colinear elements;

a frame can furthermore include the same vector multiple times.<sup>22</sup> The third of these examples is the canonical dual, which will be defined shortly.

**Inner Product Computation** Suppose  $\Phi$  is a frame for  $H$ , and  $x = \Phi\alpha$  and  $y = \Phi\beta$ . Then, just as in (1.113), we can write

$$\langle x, y \rangle = \langle \Phi\alpha, \Phi\beta \rangle = \langle G\alpha, \beta \rangle = \beta^* G\alpha,$$

where  $G = \Phi^*\Phi$  is the Gram matrix defined in (1.112). This shows how to use frame expansion coefficients to convert an inner product in  $H$  to an inner product in  $\ell^2(\mathcal{J})$ .

The key difference between the Gram matrix of a frame and the Gram matrix of a basis is that  $G$  is now not necessarily invertible. In fact, it is an invertible bounded operator if and only if the frame is a Riesz basis.

**Inverse Analysis and Synthesis** Condition (1.144b) shows that if sets  $\Phi$  and  $\tilde{\Phi}$  are a dual pair of frames, synthesis operator  $\Phi$  is a left inverse of analysis operator  $\tilde{\Phi}^*$ . As we saw before with 1-tight frames,  $\Phi$  is generally not a right inverse (hence, not an inverse) of  $\tilde{\Phi}^*$  because  $\tilde{\Phi}^*\Phi \neq I$ .

The roles of the two frames in a dual pair of frames can be reversed, so

$$x = \sum_{k \in \mathcal{K}} \langle x, \varphi_k \rangle \tilde{\varphi}_k \quad (1.145a)$$

$$= \tilde{\Phi} \Phi^* x. \quad (1.145b)$$

Thus synthesis operator  $\tilde{\Phi}$  is the left inverse of analysis operator  $\Phi^*$ . This and several other elementary properties of dual pairs of frames are established in Exercise 1.52.

**Oblique Projection** If sets  $\Phi$  and  $\tilde{\Phi}$  are a dual pair of frames, the operator  $P = \tilde{\Phi}^*\Phi$  is a projection operator. Checking idempotency of  $P$  is straightforward:

$$P^2 = (\tilde{\Phi}^*\Phi)(\tilde{\Phi}^*\Phi) = \tilde{\Phi}^*(\Phi\tilde{\Phi}^*)\Phi \stackrel{(a)}{=} \tilde{\Phi}^*I\Phi = \tilde{\Phi}^*\Phi = P, \quad (1.146)$$

where (a) follows from synthesis operator  $\Phi$  being a left inverse of analysis operator  $\tilde{\Phi}^*$ .

**Canonical Dual Frame** So far we have derived properties of a dual pair of frames without regard for how to find such a pair. Given one frame  $\Phi$ , there are many frames  $\tilde{\Phi}$  that complete a dual pair with  $\Phi$ . There is a unique choice called the *canonical dual frame*<sup>23</sup> that is important because it leads to an orthogonal projection operator on  $\ell^2(\mathcal{J})$ .

<sup>22</sup>Allowing multiplicities generalizes the concept of set to *multisets*, but we will continue to use the simpler term.

<sup>23</sup>Some authors use “dual” to mean “canonical dual.” We will *not* adopt this potentially-confusing shorthand because it obscures the possible advantages that come from flexibility in the choice of a dual.

For sets  $\Phi$  and  $\tilde{\Phi}$  to form a dual pair of frames requires the associated operators to satisfy  $\Phi\tilde{\Phi}^* = I$  on  $H$ ; see (1.144b). As established in (1.146), this makes  $P = \tilde{\Phi}^*\Phi$  a projection operator. When, in addition,  $P$  is self-adjoint, it is an *orthogonal* projection operator. Setting

$$\tilde{\Phi} = (\Phi\Phi^*)^{-1}\Phi \quad (1.147a)$$

satisfies (1.144b) and yields

$$P = \tilde{\Phi}^*\Phi = \left((\Phi\Phi^*)^{-1}\Phi\right)^*\Phi = \Phi^*(\Phi\Phi^*)^{-1}\Phi,$$

which is self-adjoint. From (1.147a), the elements of the canonical dual are

$$\tilde{\varphi}_k = (\Phi\Phi^*)^{-1}\varphi_k, \quad k \in \mathcal{J}. \quad (1.147b)$$

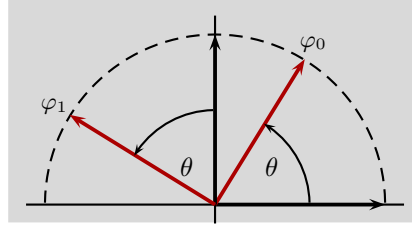
When  $H = \mathbb{C}^N$  (or  $\mathbb{R}^N$ ), the computations in (1.147) are straightforward; for example, the third dual frame in Example 1.49 is a canonical dual. In general, it is difficult to make these computations without first expressing the linear operator  $\Phi\Phi^*$  using a basis.

### 1.5.5 Matrix Representations of Vectors and Linear Operators

A basis for  $H$  creates a one-to-one correspondence between vectors in  $H$  and sequences in  $\ell^2(\mathcal{K})$ . As discussed in Section 1.5.2, an orthonormal basis preserves geometry (inner products) in this correspondence; see Figure 1.19. Even without orthonormality, using a basis is a key step toward computational feasibility because a basis allows us to do all computations with sequences. Here our intuition from finite dimensions may get in the way of appreciating what we have gained because we take the basis for granted. Computations in the Hilbert spaces  $\mathbb{C}^N$  are relatively straightforward in part because we use the standard basis automatically. Computations in other Hilbert spaces can be considerably more complicated; for example, integrating to compute an  $\mathcal{L}^2(\mathbb{R})$  inner product can be difficult. With sequences, the greatest difficulty is that if the space is infinite dimensional, the computation may require some truncation. Limiting our attention to vectors with finite  $\ell^2(\mathcal{K})$  norm ensures that the truncation can be done with small relative error; details are deferred to Chapter 5.

We get the most benefit from our experience with finite-dimensional linear algebra by thinking of sequences in  $\ell^2(\mathcal{K})$  as (possibly-infinite) column vectors. A linear operator can then be represented with ordinary matrix–vector multiplication by a (possibly-infinite) matrix. One goal in the choice of bases for the domain and codomain of the operator is to make this matrix simple. While like a basis a frame also enables representations using sequences, lack of uniqueness of the representations creates some additional intricacies; these are explored through exercises.

**Change of Basis: Orthonormal Bases** Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}}$  and  $\Psi = \{\psi_k\}_{k \in \mathcal{K}}$  be orthonormal bases for Hilbert space  $H$ . Since bases provide unique representations,



**Figure 1.27:** An orthonormal basis in  $\mathbb{R}^2$  generated by rotation of the standard basis.

for any  $x$  in  $H$  we can use synthesis operators to write  $x = \Phi\alpha$  and  $x = \Psi\beta$  for unique  $\alpha$  and  $\beta$  in  $\ell^2(\mathcal{K})$ . The operator  $C_{\Phi,\Psi} : \ell^2(\mathcal{K}) \rightarrow \ell^2(\mathcal{K})$  that maps  $\alpha$  to  $\beta$  is a *change of basis* from  $\Phi$  to  $\Psi$ .

Since  $\Psi$  has inverse  $\Psi^*$ , we could simply write  $C_{\Phi,\Psi} = \Psi^*\Phi$ . This solves our problem because

$$C_{\Phi,\Psi}\alpha = (\Psi^*\Phi)\alpha = \Psi^*(\Phi\alpha) = \Psi^*x = \beta.$$

In a finite-dimensional setting, this is a perfectly adequate solution because we know how to interpret  $\Psi^*\Phi$  as a product of matrices. We illustrate this in the following example.

**EXAMPLE 1.50 (CHANGE OF BASIS BY ROTATION)** Let  $\{\varphi_0, \varphi_1\}$  be the basis for  $\mathbb{R}^2$  shown in Figure 1.27:

$$\varphi_0 = \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix}, \quad \varphi_1 = \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix}.$$

Let  $\{\psi_0, \psi_1\}$  be the standard basis for  $\mathbb{R}^2$ . The change of basis matrix from  $\Phi$  to  $\Psi$  is

$$C_{\Phi,\Psi} = \Psi^*\Phi = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}.$$

Consider the vector in  $\mathbb{R}^2$  that has representation  $\alpha = [1 \ 0]^T$  with respect to  $\Phi$  (not with respect to the standard basis). This means that the vector is

$$x = 1 \cdot \varphi_0 + 0 \cdot \varphi_1 = \varphi_0 = \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix},$$

where the final expression is with respect to the standard basis  $\Psi$ . This agrees with the result of multiplying  $C_{\Phi,\Psi}\alpha$ .

Multiplying by  $C_{\Phi,\Psi}$  is a counterclockwise rotation by angle  $\theta$ . This agrees with the fact that the basis  $\Phi$  is the standard basis  $\Psi$  rotated counterclockwise by  $\theta$ .



In the previous example, since multiplication by a  $2 \times 2$  matrix is simple, it makes little difference whether we interpret  $\Psi^* \Phi$  as a composition of two operators or as a single operator. In general,  $C_{\Phi, \Psi}$  should not be implemented as a composition of  $\Phi$  followed by  $\Psi^*$  because we do not want to return to computations in  $H$ , which may be more complicated than computations on coefficient sequences. Instead, we would like to think of  $C_{\Phi, \Psi}$  as a  $|\mathcal{K}| \times |\mathcal{K}|$  matrix, even if  $|\mathcal{K}|$  is not finite.

Because of linearity, we can form the matrix  $C_{\Phi, \Psi}$  by finding  $C_{\Phi, \Psi} \alpha$  for particular values of  $\alpha$ . Let  $\alpha = \delta_k$  for some  $k \in \mathcal{K}$ , using the Kronecker delta notation from (1.9). Then  $x = \Phi \alpha = \varphi_k$ . Since  $\Psi$  is an orthonormal basis, the unique expansion of  $x$  with respect to  $\Psi$  is

$$x = \sum_{i \in \mathcal{K}} \langle x, \psi_i \rangle \psi_i = \sum_{i \in \mathcal{K}} \langle \varphi_k, \psi_i \rangle \psi_i,$$

from which we read off  $i$ th coefficient of  $\beta$  as  $\beta_i = \langle \varphi_k, \psi_i \rangle$ . This implies that column  $k$  of matrix  $C_{\Phi, \Psi}$  is  $\{\langle \varphi_k, \psi_i \rangle\}_{i \in \mathcal{K}}$ . The full matrix, written for the case of  $\mathcal{K} = \mathbb{Z}$ , is

$$C_{\Phi, \Psi} = \begin{bmatrix} \vdots & \vdots & \vdots \\ \cdots & \langle \varphi_{-1}, \psi_{-1} \rangle & \langle \varphi_0, \psi_{-1} \rangle & \langle \varphi_1, \psi_{-1} \rangle & \cdots \\ \cdots & \langle \varphi_{-1}, \psi_0 \rangle & \boxed{\langle \varphi_0, \psi_0 \rangle} & \langle \varphi_1, \psi_0 \rangle & \cdots \\ \cdots & \langle \varphi_{-1}, \psi_1 \rangle & \langle \varphi_0, \psi_1 \rangle & \langle \varphi_1, \psi_1 \rangle & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix}. \quad (1.148)$$

**EXAMPLE 1.51 (CHANGE TO STANDARD BASIS)** Let  $\Phi = \{\varphi_k\}_{k \in \mathbb{Z}}$  be any orthonormal basis for  $\ell^2(\mathbb{Z})$ , and let  $\Psi$  be the standard basis for  $\ell^2(\mathbb{Z})$ . Then for any integers  $k$  and  $i$ ,

$$\langle \varphi_k, \psi_i \rangle = \varphi_{k,i},$$

the  $i$ th-indexed entry of the  $k$ th vector of  $\Phi$ . The change of basis operator (1.148) simplifies to

$$C_{\Phi, \Psi} = \begin{bmatrix} \vdots & \vdots & \vdots \\ \cdots & \varphi_{-1,-1} & \varphi_{0,-1} & \varphi_{1,-1} & \cdots \\ \cdots & \varphi_{-1,0} & \boxed{\varphi_{0,0}} & \varphi_{1,0} & \cdots \\ \cdots & \varphi_{-1,1} & \varphi_{0,1} & \varphi_{1,1} & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix},$$

a matrix with the initial basis elements as columns.

**Change of Basis: Biorthogonal Pairs of Bases** We now derive the change of basis operator without assuming that the bases are orthonormal. Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}}$  and  $\Psi = \{\psi_k\}_{k \in \mathcal{K}}$  be bases for Hilbert space  $H$ . For any  $x$  in  $H$ , we can again write  $x = \Phi \alpha$  and  $x = \Psi \beta$  for unique  $\alpha$  and  $\beta$  in  $\ell^2(\mathcal{K})$ .

Since  $\Psi$  must be invertible, we could simply write  $C_{\Phi, \Psi} = \Psi^{-1} \Phi$  because then

$$C_{\Phi, \Psi} \alpha = (\Psi^{-1} \Phi) \alpha = \Psi^{-1} (\Phi \alpha) = \Psi^{-1} x = \beta.$$

As before, we would not want to implement  $C_{\Phi, \Psi}$  as a composition of two operators where the first returns computations to  $H$ . Here we have the additional complication that  $\Phi^{-1}$  may be difficult to interpret.

Because of linearity, we can form the matrix  $C_{\Phi, \Psi}$  by finding  $C_{\Phi, \Psi} \alpha$  for particular values of  $\alpha$ . Let  $\alpha = \delta_k$  for some  $k \in \mathcal{K}$ . Then  $x = \Phi \alpha = \varphi_k$ . If  $\Psi$  and  $\tilde{\Psi}$  form a biorthogonal pair of bases for  $H$ , the unique expansion of  $x$  with respect to  $\Psi$  is

$$x = \sum_{i \in \mathcal{K}} \langle x, \tilde{\psi}_i \rangle \psi_i = \sum_{i \in \mathcal{K}} \langle \varphi_k, \tilde{\psi}_i \rangle \psi_i,$$

from which we read off  $i$ th coefficient of  $\beta$  as  $\beta_i = \langle \varphi_k, \tilde{\psi}_i \rangle$ . This implies that column  $k$  of matrix  $C_{\Phi, \Psi}$  is  $\{\langle \varphi_k, \tilde{\psi}_i \rangle\}_{i \in \mathcal{K}}$ . The full matrix, written for the case of  $\mathcal{K} = \mathbb{Z}$ , is

$$C_{\Phi, \Psi} = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & \langle \varphi_{-1}, \tilde{\psi}_{-1} \rangle & \langle \varphi_0, \tilde{\psi}_{-1} \rangle & \langle \varphi_1, \tilde{\psi}_{-1} \rangle & \cdots \\ \cdots & \langle \varphi_{-1}, \tilde{\psi}_0 \rangle & \boxed{\langle \varphi_0, \tilde{\psi}_0 \rangle} & \langle \varphi_1, \tilde{\psi}_0 \rangle & \cdots \\ \cdots & \langle \varphi_{-1}, \tilde{\psi}_1 \rangle & \langle \varphi_0, \tilde{\psi}_1 \rangle & \langle \varphi_1, \tilde{\psi}_1 \rangle & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}. \quad (1.149)$$

Note that  $C_{\Phi, \Psi}$  depends on only one dual—the dual of the new representation basis  $\Psi$ . If the dual  $\tilde{\Psi}$  were not already available, computation of  $C_{\Phi, \Psi}$  could be written in terms of the inner products in (1.148) and the Gram matrix of  $\Psi$ .

**Matrix Representation of Linear Operator with Orthonormal Bases** Consider a Hilbert space  $H$  with orthonormal basis  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}}$ , and let  $A : H \rightarrow H$  be a linear operator. A matrix representation  $\Gamma$  allows  $A$  to be computed directly on coefficient sequences in the following sense: If

$$y = Ax \quad (1.150a)$$

where

$$x = \sum_{i \in \mathcal{K}} \alpha_i \varphi_i \quad (1.150b)$$

and

$$y = \sum_{k \in \mathcal{K}} \beta_k \varphi_k, \quad (1.150c)$$

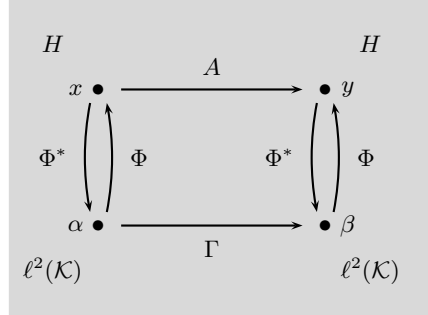
then

$$\beta = \Gamma \alpha. \quad (1.150d)$$

These relationships are depicted in Figure 1.28.

To find the matrix representation, note that the  $k$ th coefficient of the expansion of  $y$  with respect to  $\Phi$  is

$$\begin{aligned} \beta_k &\stackrel{(a)}{=} \langle y, \varphi_k \rangle \stackrel{(b)}{=} \langle Ax, \varphi_k \rangle \stackrel{(c)}{=} \langle A(\sum_{i \in \mathcal{K}} \alpha_i \varphi_i), \varphi_k \rangle \\ &\stackrel{(d)}{=} \langle \sum_{i \in \mathcal{K}} \alpha_i A\varphi_i, \varphi_k \rangle \stackrel{(e)}{=} \sum_{i \in \mathcal{K}} \alpha_i \langle A\varphi_i, \varphi_k \rangle, \end{aligned} \quad (1.151)$$



**Figure 1.28:** Conceptual illustration of the computing a linear operator  $A : H \rightarrow H$  using a matrix multiplication  $\Gamma : \ell^2(\mathcal{K}) \rightarrow \ell^2(\mathcal{K})$ . The abstract  $y = Ax$  can be replaced by the more concrete  $\beta = \Gamma\alpha$ , where  $\alpha$  and  $\beta$  are the representations of  $x$  and  $y$  with respect to orthonormal basis  $\Phi$  of  $H$ .

where (a) follows from the expression for expansion coefficients in an orthonormal basis, (1.84a); (b) from (1.150a); (c) from (1.150b); (d) from the linearity of  $A$ ; and (e) from the linearity in the first argument of the inner product. This computation of one component of  $\beta$  as a linear combination of components of  $\alpha$  determines one row of the matrix  $\Gamma$ . By gathering the equations (1.151) for all  $k \in \mathcal{K}$  (and assuming  $\mathcal{K} = \mathbb{Z}$  for concreteness), we obtain

$$\Gamma = \begin{bmatrix} \vdots & \vdots & \vdots \\ \cdots & \langle A\varphi_{-1}, \varphi_{-1} \rangle & \langle A\varphi_0, \varphi_{-1} \rangle & \langle A\varphi_1, \varphi_{-1} \rangle & \cdots \\ \cdots & \langle A\varphi_{-1}, \varphi_0 \rangle & \boxed{\langle A\varphi_0, \varphi_0 \rangle} & \langle A\varphi_1, \varphi_0 \rangle & \cdots \\ \cdots & \langle A\varphi_{-1}, \varphi_1 \rangle & \langle A\varphi_0, \varphi_1 \rangle & \langle A\varphi_1, \varphi_1 \rangle & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix}. \quad (1.152)$$

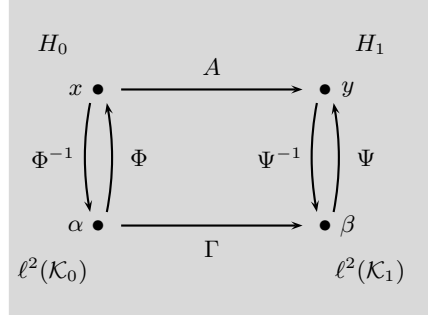
To check that (1.152) makes sense in a simple special case, let  $H = \mathbb{C}^N$  and let  $\Phi$  be the standard basis. For any  $k$  and  $i$  in  $\{0, 1, \dots, N-1\}$ ,

$$\Gamma_{i,k} = \langle A\varphi_k, \varphi_i \rangle = A_{i,k}$$

because  $A\varphi_k$  is the  $k$ th column of  $A$ , and taking the inner product with  $\varphi_i$  picks out the  $i$ th entry. Thus the conventional use of matrices for linear operators on  $\mathbb{C}^N$  is consistent with (1.152). This extends also to the use of the standard basis of  $\ell^2(\mathbb{Z})$ .

For a given operator  $A$ , a frequent goal in choosing the basis is to make  $\Gamma$  simple, for example, diagonal.

**EXAMPLE 1.52 (DIAGONALIZING BASIS)** Let  $H = \mathbb{R}^N$ , and consider a linear operator  $A : H \rightarrow H$  given by a symmetric matrix. Such a matrix can be decomposed as  $A = \Phi\Lambda\Phi^T$ , where the columns of unitary matrix  $\Phi$  are eigenvectors of  $A$  and  $\Lambda$  is the diagonal matrix of corresponding eigenvalues,  $\{\lambda_0, \lambda_1, \dots, \lambda_{N-1}\}$ ;



**Figure 1.29:** Conceptual illustration of the computing a linear operator  $A : H_0 \rightarrow H_1$  using a matrix multiplication  $\Gamma : \ell^2(\mathcal{K}_0) \rightarrow \ell^2(\mathcal{K}_1)$ . The abstract  $y = Ax$  can be replaced by the more concrete  $\beta = \Gamma\alpha$ , where  $\alpha$  is the representation of  $x$  with respect to basis  $\Phi$  of  $H_0$ , and  $\beta$  is the representation of  $y$  with respect to basis  $\Psi$  of  $H_1$ .

see (1.223b). Expressing the operator with respect to the orthonormal basis  $\Phi$  we obtain

$$\Gamma_{i,k} \stackrel{(a)}{=} \langle A\varphi_k, \varphi_i \rangle \stackrel{(b)}{=} \langle \lambda_k \varphi_k, \varphi_i \rangle \stackrel{(c)}{=} \lambda_k \langle \varphi_k, \varphi_i \rangle \stackrel{(d)}{=} \lambda_k \delta_{i-k},$$

where (a) follows from (1.152); (b) from  $(\lambda_k, \varphi_k)$  being an eigenpair of  $A$ ; (c) from linearity in the first argument of the inner product; and (d) from the orthonormality of  $\Phi$ . Thus, the representation is diagonal: multiplication of a vector by a matrix  $A$  is replaced by pointwise multiplication of expansion coefficients of  $x$  by the eigenvalues of  $A$ .

This simple example is fundamental since many basis changes aim to diagonalize operators. For example, we will see in Chapter 2 that the discrete Fourier transform diagonalizes the circular convolution operator because it is formed from the eigenvectors of the circular convolution operator. As in the example, multiplication by a dense matrix becomes a pointwise multiplication in a new basis.

Moving to cases where the domain and codomain of the linear operator are not necessarily the same Hilbert space, consider a linear operator  $A : H_0 \rightarrow H_1$ , and let  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}_0}$  be an orthonormal basis for  $H_0$  and  $\Psi = \{\psi_k\}_{k \in \mathcal{K}_1}$  be an orthonormal basis for  $H_1$ . We would like to implement  $A$  as an operation on sequence representations with respect to  $\Phi$  and  $\Psi$ . The concept is depicted in Figure 1.29, where orthonormality of the bases gives  $\Phi^{-1} = \Phi^*$  and  $\Psi^{-1} = \Psi^*$ . The computation  $y = Ax$  is replaced by  $\beta = \Gamma\alpha$ , where  $\alpha$  is the representation of  $x$  with respect to  $\Phi$  and  $\beta$  is the representation of  $y$  with respect to  $\Psi$ .

Mimicking the derivation of (1.151) leads to a counterpart for (1.152) that

## 1.5. Bases and Frames

109

uses both bases:

$$\Gamma = \begin{bmatrix} \vdots & \vdots & \vdots \\ \cdots & \langle A\varphi_{-1}, \psi_{-1} \rangle & \langle A\varphi_0, \psi_{-1} \rangle & \langle A\varphi_1, \psi_{-1} \rangle & \cdots \\ \cdots & \langle A\varphi_{-1}, \psi_0 \rangle & \boxed{\langle A\varphi_0, \psi_0 \rangle} & \langle A\varphi_1, \psi_0 \rangle & \cdots \\ \cdots & \langle A\varphi_{-1}, \psi_1 \rangle & \langle A\varphi_0, \psi_1 \rangle & \langle A\varphi_1, \psi_1 \rangle & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix}. \quad (1.153)$$

Note the information upon which  $\Gamma$  is determined: by linearity of  $A$  and completeness of the basis  $\Phi$ , the effect of  $A$  on any  $x \in H_0$  can be computed from its effect on each basis element of the domain space,  $\{A\varphi_k\}_{k \in \mathcal{K}_0}$ ; and the expansion coefficients of any one of these results is determined by inner products with the basis in the codomain space,  $\{\langle A\varphi_k, \psi_i \rangle\}_{i \in \mathcal{K}_1}$ .

**EXAMPLE 1.53 (AVERAGING OPERATOR)** Consider the operator  $A : H_0 \rightarrow H_1$  that replaces a function by its average over intervals of length 2,

$$y(t) = Ax(t) = \frac{1}{2} \int_{2\ell}^{2(\ell+1)} x(\tau) d\tau, \quad \text{for } 2\ell \leq t < 2(\ell+1), \quad \ell \in \mathbb{Z}, \quad (1.154)$$

where  $H_0$  the space of piecewise-constant, finite-energy functions with breakpoints at integers and  $H_1$  the space of piecewise-constant, finite-energy functions with breakpoints at even integers. As orthonormal bases for  $H_0$  and  $H_1$ , we choose normalized indicator functions over unit and double-unit intervals, respectively:

$$\begin{aligned} \Phi &= \{\varphi_k(t)\}_{k \in \mathbb{Z}} = \{\chi_{[k, k+1)}(t)\}_{k \in \mathbb{Z}}, \\ \Psi &= \{\psi_i(t)\}_{i \in \mathbb{Z}} = \left\{ \frac{1}{\sqrt{2}} \chi_{[2i, 2(i+1))}(t) \right\}_{i \in \mathbb{Z}}, \end{aligned}$$

with the indicator function defined as

$$\chi_I(t) = \begin{cases} 1, & \text{for } t \in I; \\ 0, & \text{otherwise.} \end{cases} \quad (1.155)$$

To evaluate  $\Gamma$  from (1.153) requires  $\langle A\varphi_k, \psi_i \rangle$  for all integers  $k$  and  $i$ . Since  $\varphi_0(t)$  is nonzero only for  $t \in [0, 1)$ ,

$$A\varphi_0(t) = \begin{cases} \frac{1}{2}, & \text{for } 0 \leq t < 2; \\ 0, & \text{otherwise,} \end{cases}$$

from which

$$\langle A\varphi_0, \psi_0 \rangle = \int_0^2 \frac{1}{2} \frac{1}{\sqrt{2}} d\tau = \frac{1}{\sqrt{2}}$$

and

$$\langle A\varphi_0, \psi_i \rangle = 0 \quad \text{for all } i \neq 0.$$

Since  $A$  integrates over intervals of the form  $[2\ell, 2(\ell+1)]$ ,  $A\varphi_1 = A\varphi_0$ , so

$$\langle A\varphi_1, \psi_i \rangle = \begin{cases} \frac{1}{\sqrt{2}}, & \text{for } i = 0; \\ 0, & \text{otherwise.} \end{cases}$$

Continuing the computation to cover every  $\varphi_k$  yields  $\Gamma$ :

$$\Gamma = \frac{1}{\sqrt{2}} \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & 1 & 1 & 0 & 0 & 0 & 0 & \cdots \\ \cdots & 0 & 0 & \boxed{1} & 1 & 0 & 0 & \cdots \\ \cdots & 0 & 0 & 0 & 0 & 1 & 1 & \cdots \\ & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \quad (1.156)$$

Multiplying by  $\Gamma$  is thus a very simple operation.

**Matrix Representation of Linear Operator with Biorthogonal Pairs of Bases** As above, consider a linear operator  $A : H_0 \rightarrow H_1$ . Assume  $\Phi$  and  $\tilde{\Phi}$  form a biorthogonal pair of bases for  $H_0$ , and  $\Psi$  and  $\tilde{\Psi}$  form a biorthogonal pair of bases for  $H_1$ . We would like to implement  $A$  as an operation on sequence representations with respect to  $\Phi$  and  $\Psi$  as in Figure 1.29, where biorthogonality of the bases gives  $\Phi^{-1} = \tilde{\Phi}^*$  and  $\Psi^{-1} = \tilde{\Psi}^*$ .

Derivation of  $\Gamma$ , the matrix representation of the operator  $A$ , is almost unchanged from orthonormal case, but we repeat the key computation to show the role of having biorthogonal bases. When

$$x = \sum_{i \in \mathcal{K}_0} \alpha_i \varphi_i, \quad (1.157)$$

the expansion of

$$y = Ax \quad (1.158)$$

with respect to  $\Psi$  has  $k$ th coefficient

$$\begin{aligned} \beta_k &\stackrel{(a)}{=} \langle y, \tilde{\psi}_k \rangle \stackrel{(b)}{=} \langle Ax, \tilde{\psi}_k \rangle \stackrel{(c)}{=} \langle A(\sum_{i \in \mathcal{K}_0} \alpha_i \varphi_i), \tilde{\psi}_k \rangle \\ &\stackrel{(d)}{=} \langle \sum_{i \in \mathcal{K}_0} \alpha_i A\varphi_i, \tilde{\psi}_k \rangle \stackrel{(e)}{=} \sum_{i \in \mathcal{K}_0} \alpha_i \langle A\varphi_i, \tilde{\psi}_k \rangle, \end{aligned}$$

where (a) follows from the expression for expansion coefficients with a biorthogonal pair of bases, (1.104a); (b) from (1.158); (c) from (1.157); (d) from the linearity of  $A$ ; and (e) from the linearity in the first argument of the inner product. Thus the matrix representation (and assuming  $\mathcal{K}_0 = \mathcal{K}_1 = \mathbb{Z}$  for concreteness) is

$$\Gamma = \begin{bmatrix} \vdots & \vdots & \vdots \\ \cdots & \langle A\varphi_{-1}, \tilde{\psi}_{-1} \rangle & \langle A\varphi_0, \tilde{\psi}_{-1} \rangle & \langle A\varphi_1, \tilde{\psi}_{-1} \rangle & \cdots \\ \cdots & \langle A\varphi_{-1}, \tilde{\psi}_0 \rangle & \boxed{\langle A\varphi_0, \tilde{\psi}_0 \rangle} & \langle A\varphi_1, \tilde{\psi}_0 \rangle & \cdots \\ \cdots & \langle A\varphi_{-1}, \tilde{\psi}_1 \rangle & \langle A\varphi_0, \tilde{\psi}_1 \rangle & \langle A\varphi_1, \tilde{\psi}_1 \rangle & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix}. \quad (1.159)$$

Comparing (1.153) and (1.159), the only difference is the use of the dual basis for the codomain space; this is natural since we require expansions of  $\{A\varphi_k\}_{k \in \mathcal{K}_0}$  with respect to  $\Psi$ . Also note the similarity between (1.149) and (1.159); a change of basis operator is a special case of a matrix representation of a linear operator where  $H_0 = H_1 = H$  and the operator is the identity on  $H$ .

The next example points to how differential operators can be implemented as matrix multiplications once bases for the domain and codomain spaces are available.

**EXAMPLE 1.54 (DERIVATIVE OPERATOR)** Consider the derivative operator  $A : H_0 \rightarrow H_1$ , with  $H_0$  the space of piecewise-linear, continuous, finite-energy functions with breakpoints at integers and  $H_1$  the space of piecewise-constant, finite-energy functions with breakpoints at integers. As a basis for  $H_0$ , choose the hat function

$$\varphi(t) = \begin{cases} 1 - |t|, & \text{for } |t| < 1; \\ 0, & \text{otherwise} \end{cases} \quad (1.160)$$

and its integer shifts:

$$\Phi = \{\varphi_k(t)\}_{k \in \mathbb{Z}} = \{\varphi(t - k)\}_{k \in \mathbb{Z}}.$$

(This is an infinite-dimensional analogue to the basis in Example 1.41 and an example of a spline, discussed in detail in Chapter 5.) For  $H_1$ , we can choose the same orthonormal basis as in Example 1.53:

$$\Psi = \{\psi_i(t)\}_{i \in \mathbb{Z}} = \{\chi_{[i, i+1)}(t)\}_{i \in \mathbb{Z}}.$$

To evaluate  $\Gamma$  from (1.159) requires  $\langle A\varphi_k, \tilde{\psi}_i \rangle$  for all integers  $k$  and  $i$ . Since

$$A\varphi(t) = \varphi'(t) = \begin{cases} 1, & \text{for } -1 < t < 0; \\ -1, & \text{for } 0 < t < 1; \\ 0, & \text{for } |t| > 1, \end{cases}$$

it follows that

$$\langle A\varphi_0, \tilde{\psi}_i \rangle = \begin{cases} 1, & \text{for } i = -1; \\ -1, & \text{for } i = 0; \\ 0, & \text{otherwise.} \end{cases}$$

Shifting  $\varphi(t)$  by  $k$  simply shifts the derivative:

$$\langle A\varphi_k, \tilde{\psi}_i \rangle = \begin{cases} 1, & \text{for } i = k - 1; \\ -1, & \text{for } i = k; \\ 0, & \text{otherwise.} \end{cases}$$

Gathering these computations into a matrix yields

$$\Gamma = \begin{bmatrix} \ddots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots & \\ \cdots & 0 & -1 & 1 & 0 & 0 & 0 & \cdots \\ \cdots & 0 & 0 & \boxed{-1} & 1 & 0 & 0 & \cdots \\ \cdots & 0 & 0 & 0 & -1 & 1 & 0 & \cdots \\ & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix}. \quad (1.161)$$

Figure 1.30 gives an example of a derivative operator and its computation. The input function and expansion coefficients in the basis  $\Phi$  are

$$x(t) = \varphi(t) - \varphi(t-1),$$

$$\alpha = \left[ \dots \quad 0 \quad \boxed{1} \quad -1 \quad 0 \quad 0 \quad \dots \right]^T,$$

while its derivative and expansion coefficients in the basis  $\Psi$  are

$$x'(t) = \psi(t+1) - 2\psi(t) + \psi(t-1),$$

$$\beta = \left[ \dots \quad 0 \quad 1 \quad -2 \quad 1 \quad 0 \quad \dots \right]^T.$$

Then, indeed

$$\beta = \begin{bmatrix} \vdots \\ 0 \\ 1 \\ \boxed{-2} \\ 1 \\ 0 \\ \vdots \end{bmatrix} = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & 0 & -1 & 1 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & \boxed{-1} & 1 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & -1 & 1 & 0 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ 0 \\ 0 \\ \boxed{1} \\ -1 \\ 0 \\ 0 \\ \vdots \end{bmatrix} = \Gamma \alpha.$$

**Matrix Representation of Adjoint** Example 1.15(ii) confirmed that the adjoint of a linear operator  $A : \mathbb{C}^N \rightarrow \mathbb{C}^M$  given by a finite matrix is the Hermitian transpose of the matrix; implicit in this was the use of the standard bases for  $\mathbb{C}^N$  and  $\mathbb{C}^M$ . The connection between the adjoint and the Hermitian transpose of a matrix extends to arbitrary Hilbert spaces and linear operators when orthonormal bases are used.

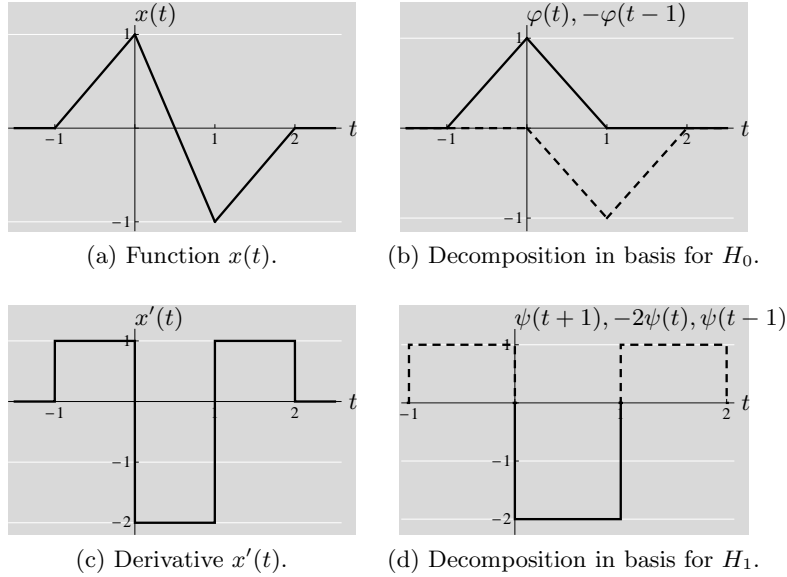
Consider a linear operator  $A : H_0 \rightarrow H_1$ , and let  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}_0}$  be an orthonormal basis for  $H_0$  and  $\Psi = \{\psi_k\}_{k \in \mathcal{K}_1}$  be an orthonormal basis for  $H_1$ . Let  $\Gamma$  be the matrix representation of  $A$  with respect to  $\Phi$  and  $\Psi$ , as given by (1.153). The adjoint  $A^*$  is an operator  $H_1 \rightarrow H_0$ , and we would like to find its matrix representation with respect to  $\Psi$  and  $\Phi$ . Applying (1.153) to  $A^*$ , the entry in row  $i$ , column  $k$  is

$$\langle A^* \psi_k, \varphi_i \rangle \stackrel{(a)}{=} \langle \psi_k, A \varphi_i \rangle \stackrel{(b)}{=} \langle A \varphi_i, \psi_k \rangle^* \stackrel{(c)}{=} \Gamma_{k,i}^*,$$

where (a) follows from the definition of adjoint; (b) from Hermitian symmetry of the inner product; and (c) from (1.153). Thus, the matrix representation of  $A^*$  is indeed the Hermitian transpose of the matrix representation of  $A$ .

Now remove the assumption that  $\Phi$  and  $\Psi$  are orthonormal bases and denote the respective dual bases by  $\tilde{\Phi}$  and  $\tilde{\Psi}$ . Let  $\Gamma$  be the matrix representation of  $A$  with respect to  $\Phi$  and  $\Psi$ , as given by (1.159). The matrix representation of  $A^*$  has a



**Figure 1.30:** Example of derivative operator.

simple form with respect to the *duals*  $\tilde{\Psi}$  for  $H_1$  and  $\tilde{\Phi}$  for  $H_0$ . Applying (1.159) to  $A^*$ , the entry in row  $i$ , column  $k$  is

$$\langle A^* \tilde{\psi}_k, \varphi_i \rangle \stackrel{(a)}{=} \langle \tilde{\psi}_k, A\varphi_i \rangle \stackrel{(b)}{=} \langle A\varphi_i, \tilde{\psi}_k \rangle^* \stackrel{(c)}{=} \Gamma_{k,i}^*,$$

where (a) follows from the definition of adjoint; (b) from Hermitian symmetry of the inner product; and (c) from (1.159). Thus, the matrix representation of  $A^*$  is the Hermitian transpose of the matrix representation of  $A$  *when the bases are switched to the duals*. To represent  $A^*$  with respect to  $\Psi$  and  $\Phi$  rather than with respect to their duals is a bit more complicated.

## 1.6 Computational Aspects

The cost of an algorithm is generally measured by the number of operations needed and the precision requirements for both the input data and the intermediate results. These cost metrics are of primary interest and enable comparisons that are independent of the computation platform. Running time and hardware resources (such as chip area) can be traded off through parallelization and are also affected by more subtle algorithmic properties.

We start with the basics of using operation counts to express complexity of a problem and cost of an algorithm. We then discuss precision of the computation in terms of the arithmetic representation of a number, followed by conditioning as the sensitivity of the solution to changes in the data. We close with one of the most fundamental problems in linear algebra: solving systems of linear equations.

### 1.6.1 Cost, Complexity, and Asymptotic Notations

**Cost and Complexity** For a given problem, many algorithms may exist. We can measure the *cost* of computing each of these algorithms and define the *complexity* of the problem as the minimum cost over any possible algorithm. The definition of cost should reflect the consumption of relevant resources—computation time, memory, circuit area, energy, etc. Sometimes these resources can themselves be traded off; for example, parallelism trades area for time or, since slower circuits can operate at lower power, area for energy. These trade-offs depend on intricacies of hardware implementations, but counting arithmetic operations is enough for high-level comparisons of algorithms. In particular, we will see benefits from certain problem structures. Traditionally, one counts multiplications or both multiplications and additions. A multiplication is typically more expensive than an addition, as can be seen from the steps involved in long-hand multiplication of binary numbers.

The complexity of a problem depends on the computational model (which operations are allowed and their costs), the possible inputs, and the format of the input.

EXAMPLE 1.55 (COMPLEXITY OF POLYNOMIAL EVALUATION) There are several algorithms to evaluate

$$x(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3. \quad (1.162)$$

The most obvious is through

$$\text{output : } a_0 + a_1 \cdot t + (a_2 \cdot t) \cdot t + ((a_3 \cdot t) \cdot t) \cdot t,$$

which has  $\mu = 6$  multiplications and  $\nu = 3$  additions. This is wasteful because powers of  $t$  could have been saved and reused. Specifically, the computations

$$t_2 = t \cdot t; \quad t_3 = t_2 \cdot t; \quad \text{output : } a_0 + a_1 \cdot t + a_2 \cdot t_2 + a_3 \cdot t_3$$

give the same final result with  $\mu = 5$  and  $\nu = 3$ . An even cheaper algorithm is

$$\text{output : } a_0 + t \cdot (a_1 + t \cdot (a_2 + t \cdot a_3)),$$

with  $\mu = 3$  and  $\nu = 3$ . In fact,  $\mu = 3$  and  $\nu = 3$  are the minimum possible multiplicative and additive costs (and hence the problem complexity) for *arbitrary* input  $(a_0, a_1, a_2, a_3, t)$ . Restrictions on the input could reduce the complexity.

Other formats for the same polynomial lead to different algorithms with different costs. For example, if the polynomial is given in its factored form

$$x(t) = b_0(t + b_1)(t + b_2)(t + b_3), \quad (1.163)$$

it will have a natural implementation with  $\mu = 3$  and  $\nu = 3$ , matching the complexity of the problem. However, a real polynomial could have complex roots. When using real operations to measure cost, one could assign a complex multiplication  $(\mu, \nu) = (4, 2)$  and a complex addition  $(\mu, \nu) = (0, 2)$ .<sup>24</sup> The

<sup>24</sup>These costs are based on the obvious implementation of complex multiplication as  $(a + jb)(c + jd) = (ac - bd) + j(ad + bc)$ ; a complex multiplication could be computed with 3 real multiplications and 5 real additions, as in Exercise 1.59.

algorithm based on the factored form (1.163) then has higher cost than that based on the expanded form (1.162).

This example illustrates that mathematically-equivalent expressions may not be equivalent for computation. We will revisit this, again in the context of polynomials, when we discuss precision.

The scaling of cost and complexity with the problem size is typically of interest. For example, the complexity determined in Example 1.55 generalizes to  $\mu$  and  $\nu$  equaling the degree of the polynomial. Finding exact complexities is usually very difficult, and we are satisfied with coarse descriptions expressed with asymptotic notations.

**Asymptotic Notation** The most common asymptotic notation is the *big O*, rooted in the word *order*. While we define it and other asymptotic notations for sequences indexed by  $n \in \mathbb{N}$  with  $n \rightarrow \infty$ , the same notation is used for functions with the argument approaching any finite or infinite limit. Informally,  $x = O(y)$  means that  $x_n$  is eventually (for large enough  $n$ ) bounded from above by a constant multiple of  $y_n$ .

**DEFINITION 1.53 (ASYMPTOTIC NOTATION)** Let  $x$  and  $y$  be sequences defined on  $\mathbb{N}$ . We say:

- (i)  $x$  is  $O$  of  $y$  and write  $x \in O(y)$  or  $x = O(y)$  when there exist constants  $\gamma > 0$  and  $n_0 \in \mathbb{N}$  such that

$$0 \leq x_n \leq \gamma y_n, \quad \text{for all } n \geq n_0; \quad (1.164)$$

- (ii)  $x$  is  $o$  of  $y$  and write  $x \in o(y)$  or  $x = o(y)$  when, for any  $\gamma > 0$ , there exists a constant  $n_0 \in \mathbb{N}$  such that (1.164) holds;
- (iii)  $x$  is  $\Omega$  of  $y$  and write  $x \in \Omega(y)$  or  $x = \Omega(y)$  when there exist constants  $\gamma > 0$  and  $n_0 \in \mathbb{N}$  such that

$$\gamma y_n \leq x_n, \quad \text{for all } n \geq n_0;$$

- (iv)  $x$  is  $\Theta$  of  $y$  and write  $x \in \Theta(y)$  or  $x = \Theta(y)$  when  $x$  is both  $O$  of  $y$  and  $\Omega$  of  $y$ , that is  $x \in O(y) \cap \Omega(y)$ .

The convenient asymptotic notations necessitate a few notes of caution: The use of an equals sign is an abuse of that symbol because asymptotic notations are not symmetric relations. Also, if the argument over which one is taking a limit is not clear from the context, it should be written explicitly; for example,  $2^m n^2 = O_n(n^2)$  is correct when  $m$  does not depend on  $n$  because the added subscript specifies that we are interested only in the scaling with respect to  $n$ . Finally, all the asymptotic notations omit constant factors that can be critical in assessing and comparing algorithms.

**Computing the Cost of an Algorithm** Most often, we will compute the cost of an algorithm in terms of the number of operations, *multiplications*  $\mu$  and *additions*  $\nu$ , for a total cost of

$$C = \mu + \nu, \quad (1.165)$$

followed by an asymptotic estimation of its behavior in  $O$  notation.

**EXAMPLE 1.56 (MATRIX MULTIPLICATION)** We illustrate cost and complexity with one of the most basic operations in linear algebra: matrix multiplication. Using the definition (1.204) directly, the product  $Q = AB$ , with  $A \in \mathbb{C}^{M \times N}$  and  $B \in \mathbb{C}^{N \times P}$ , requires  $N$  multiplications and  $N - 1$  additions for each  $Q_{ik}$ , for a total of  $\mu = MNP$  multiplications and  $\nu = M(N - 1)P$  additions, and a total cost of  $C = M(2N - 1)P$ . Setting  $M = P = N$ , the cost for multiplying two  $N \times N$  matrices is

$$C_{\text{mat-mat}} = 2N^3 - N^2; \quad (1.166a)$$

and setting  $M = N$  and  $P = 1$ , the cost of multiplying an  $N \times N$  matrix by an  $N \times 1$  vector is

$$C_{\text{mat-vec}} = 2N^2 - N. \quad (1.166b)$$

By specifying that (1.204) is to be used, we have implicitly identified a particular algorithm for matrix multiplication. Other algorithms may be preferable. There are algorithms that reduce the number of multiplications at the expense of having more additions; for example, for the multiplication of  $2 \times 2$  matrices, (1.204) gives a cost of 8 multiplications, but we now show that the computation can be accomplished with 7 multiplications. The product

$$\begin{bmatrix} Q_{00} & Q_{01} \\ Q_{10} & Q_{11} \end{bmatrix} = \begin{bmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{bmatrix} \begin{bmatrix} B_{00} & B_{01} \\ B_{10} & B_{11} \end{bmatrix}$$

can be computed from intermediate results

$$\begin{aligned} h_0 &= (A_{01} - A_{11})(B_{10} + B_{11}), & h_4 &= A_{00}(B_{01} + B_{11}), \\ h_1 &= (A_{00} + A_{11})(B_{00} + B_{11}), & h_5 &= A_{11}(B_{10} + B_{00}), \\ h_2 &= (A_{00} - A_{10})(B_{00} + B_{01}), & h_6 &= (A_{10} + A_{11})B_{00}, \\ h_3 &= (A_{00} + A_{01})B_{11}, \end{aligned}$$

as

$$\begin{aligned} Q_{00} &= h_0 + h_1 - h_3 + h_5, & Q_{01} &= h_3 + h_4, \\ Q_{10} &= h_5 + h_6, & Q_{11} &= h_1 - h_2 + h_4 - h_6. \end{aligned}$$

The  $\mu = 7$  multiplications is the minimum possible, but the number of additions is increased to  $\nu = 18$ . This procedure is *Strassen's algorithm* (see Solved Exercise 1.6).

### 1.6.2 Precision

Counting operations is important, but so is the trade-off between the cost of a specific operation and its precision. In digital computation, operations are over a (possibly huge) finite set of values. Not all operations are possible in the sense that many operations will have results outside the finite set. How these cases are handled—primarily through rounding and saturation—introduces inaccuracy into computations. Properties of this inaccuracy depend heavily on the specific arithmetic representation of a number. We assume binary arithmetic and discuss two dominant cases: fixed-point and floating-point arithmetic.

**Fixed-Point Arithmetic** Fixed-point arithmetic deals with operations over a finite set of evenly-spaced values. Consider the values to be the integers between 0 and  $2^B - 1$ , where  $B$  is the number of bits used in the representation; then, a binary string  $(b_0, b_1, \dots, b_{B-1})$  defines an integer in that range, given by

$$x = b_0 + b_1 \cdot 2 + b_2 \cdot 2^2 + \dots + b_{B-1} \cdot 2^{B-1}. \quad (1.167)$$

Negative numbers are handled with an extra bit and various formats; this does not change the basic issues.

The sum of two  $B$ -bit numbers is in  $\{0, 1, \dots, 2^B - 2\}$ . The result can thus be represented exactly unless there is *overflow*, the case of the result being too large for the number format. Overflow could result in an error, saturation (setting the result to the largest number  $2^B - 1$ ), or wraparound (returning the remainder in dividing the correct result by  $2^B$ ). All of these make an algorithm difficult to analyze and are generally to be avoided. One could guarantee there is no overflow in the computation  $x + y$  by limiting  $x$  and  $y$  to the lower half of the valid numbers (requiring the most significant bit  $b_{B-1}$  to be 0). Reducing the range of possible inputs to ensure accuracy of the computation has its limitations; for example, to avoid overflow in the product  $x \cdot y$  requires restricting  $x$  and  $y$  to be less than  $2^{B/2}$  (requiring half of the input bits to be zero).

Many other operations, like square root or division by a nonzero number, have results that cannot be represented exactly. Results of these operations will generally be rounded to the nearest valid representation point. Thus, assuming no overload, the error from a fixed-point computation is bounded through

$$|x - \hat{x}| / (x_{\max} - x_{\min}) \leq \frac{1}{2} / (2^B - 1) \approx 2^{-(B+1)}, \quad (1.168)$$

where  $x$  is the exact result,  $\hat{x}$  is the result of the fixed-point computation, and  $x_{\max} - x_{\min}$  is the full range of valid representation points. The error bound is written with normalization by the range to highlight the role of the number of bits  $B$ . To contrast with what we will see for floating-point arithmetic, note that the error could be small relative to  $x$  (if  $|x|$  is large) or large relative to  $x$  (if  $x$  is near 0).

**Floating-Point Arithmetic** Floating-point representations use numbers spread over a vastly larger range so that overload is largely avoided. A floating point representation has significant digits and an exponent. The significant digits are called the

*mantissa*, or *significand*, and are defined like a fixed-point number, with the fraction point typically after the most significant digit, which is 1 by definition. The exponent is a fixed-point binary number used to scale the significand by a power of 2. Consider just strictly positive numbers and suppose  $B$  bits are divided into  $B_S$  bits for significand and  $B_E = B - B_S$  bits for exponent. Then, a number  $x$  can be represented in floating-point binary form as

$$x = \left(1 + \sum_{n=1}^{B_S} b_n 2^{-n}\right) 2^E, \quad (1.169)$$

where  $E$  is a fixed-point binary number having  $B_E$  bits chosen so that the leading bit of the significand is 1. Of course, this representation still covers a finite range of possible numbers, but because of the significand/exponent decomposition, this range is larger than compared to fixed-point arithmetic.

**EXAMPLE 1.57 (32-BIT ARITHMETIC)** In the IEEE 754-2008 standard for 32-bit floating-point arithmetic, 1 bit is reserved for the sign, 8 for the exponent, and 23 for the significand. Since the leading 1 to the left of the binary point is assumed, the significand effectively has 24 bits. The value of the number is given by

$$x = (-1)^{\text{sign}} \left(1 + \sum_{n=1}^{23} b_n 2^{-n}\right) 2^{E-127} \quad (1.170)$$

for  $E \in \{1, 2, \dots, 254\}$ . The two remaining values of  $E$  are used differently:  $E = 255$  is used for  $\pm\infty$  and “not a number” (NaN); and  $E = 0$  is used for 0 if the significand is zero and *subnormal* numbers if the significand is nonzero. The subnormal numbers extend the range of representable positive numbers below  $2^{-126}$  through

$$x = (-1)^{\text{sign}} \left(\sum_{n=1}^{23} b_n 2^{-n}\right) 2^{-126}.$$

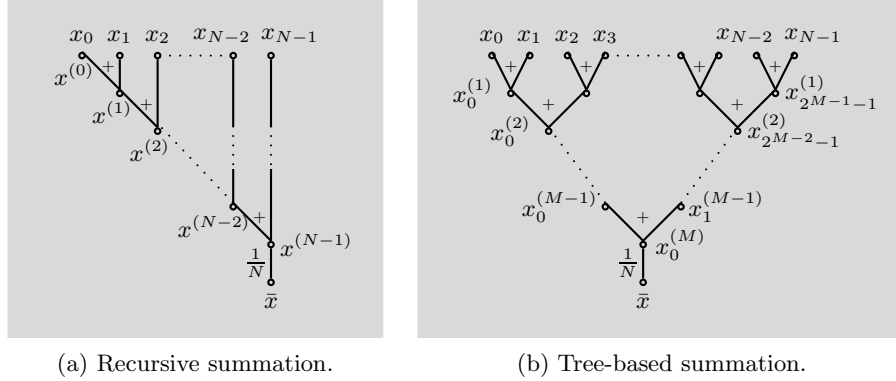
All the positive numbers that can be represented through (1.170) lie in

$$[2^{-126}, 2 \cdot 2^{127}] \approx [1.18 \cdot 10^{-38}, 3.4 \cdot 10^{38}],$$

and similarly for negative numbers. To this we add subnormal numbers, the minimum of which is approximately  $1.4 \cdot 10^{-45}$ . Comparing this to  $[0, 4.3 \cdot 10^9]$  (with integer spacing) for 32-bit fixed-point arithmetic shows an advantage of floating-point arithmetic.

In floating-point arithmetic following (1.169), the difference between a real number and the closest valid representation may be large—but only if the number itself is large. Suppose  $x$  is positive and not too large to be represented. Then its representation  $\hat{x}$  will satisfy

$$\hat{x} = (1 + \epsilon)x, \quad \text{where } |\epsilon| < 2^{-B_S}. \quad (1.171)$$

**Figure 1.31:** Two algorithms for computing an average.

This is very different than (1.168); the error  $|x - \hat{x}|$  may be large, but it is not large relative to  $x$ .

Multiplication of floating-point numbers amounts to the product of the significands and the sum of the exponents, followed possibly by rescaling. Addition is more involved, since when the exponents are not equal, the significands have to adjust the fractional point so they can be added. When the exponents are very different, a smaller term will lose precision, and could even be set to zero. Nevertheless, adding positive numbers or multiplying positive numbers, within the range of the number system, will have error satisfying (1.171). Much more troublesome is that subtracting numbers that are nearly equal can result in cancellation of many leading bits. This is called a *loss of significance* error.

**EXAMPLE 1.58 (COMPUTING AN AVERAGE)** We now highlight how the choice of an algorithm affects precision on one of the simplest operations, computation of the average of  $N$  numbers,

$$\bar{x} = \frac{1}{N} \sum_{n=0}^{N-1} x_n.$$

An obvious algorithm is the recursive procedure illustrated in Figure 1.31(a),

$$\begin{aligned} x^{(0)} &= x_0, \\ x^{(n)} &= x^{(n-1)} + x_n, \quad n = 1, 2, \dots, N, \\ \bar{x} &= x^{(N)} = \frac{1}{N} x^{(N-1)}. \end{aligned}$$

When all  $x_k$ s are close in value, the summands in step  $n$  above differ in size by a factor of  $n$  (since  $x^{(n-1)}$  is the partial sum of the first  $n$  numbers). With  $N$  large, this becomes problematic as  $n$  grows.

A simple alternative is summing on a tree as in Figure 1.31(b). Assume  $N = 2^M$ , and introduce sequences  $x_n^{(i)}$  as partial sums of  $2^i$  terms,

$$\begin{aligned} x_n^{(0)} &= x_n, \\ x_n^{(i)} &= x_{2n}^{(i-1)} + x_{2n+1}^{(i-1)}, & i = 1, 2, \dots, M, \\ & & n = 0, 1, \dots, 2^{M-i} - 1, \\ \bar{x} &= \frac{1}{N} x_0^{(M)}. \end{aligned}$$

Because all summations are of terms of similar size, the precision of the result will improve. Note that the number of additions is the same as in the previous algorithm, that is,  $N - 1$ .

### 1.6.3 Conditioning

So far, we have discussed two issues: algorithmic efficiency, or the number of operations required to solve a given problem, and precision of the computation, linked both to machine precision as well as algorithmic structure (as in Example 1.58).

We now discuss conditioning of a problem, which describes the sensitivity of the solution to changes in the data. In an ill-conditioned problem, the solution can vary widely with small changes in the input. Ill-conditioned problems also tend to be more sensitive to algorithmic choices that would be immaterial with exact arithmetic. We study conditioning by looking at the solution of systems of linear equations.

Given a system of linear equations  $y = Ax$ , where  $x$  is a set of  $N$  numbers and  $A$  is an  $N \times N$  matrix of full rank, we know a unique solution exists,  $x = A^{-1}y$ . The condition number we introduce shortly will roughly say how sensitive the solution  $x$  will be to small changes in  $y$ . In particular, if the condition number is large, a tiny change (error) in  $y$  can lead to a large change (error) in  $x$ . Conversely, a small condition number signifies that the error in  $x$  will be of the same order as the error in  $y$ .

For this discussion, we use the 2-norm as defined in (1.216):

$$\|A\|_2 = \|A\|_{2,2} = \sup_{\|x\|_2=1} \|Ax\|_2 = \sigma_{\max}(A) = \sqrt{\lambda_{\max}(A^*A)}. \quad (1.174a)$$

Similarly,

$$\|A^{-1}\|_2 = \frac{1}{\sigma_{\min}(A)} = \frac{1}{\sqrt{\lambda_{\min}(A^*A)}}. \quad (1.174b)$$

For now, consider the matrix  $A$  to be exact, and let us see how changes in  $y$ , expressed as  $\hat{y} = y + \Delta y$ , affect the changes in the solution  $x$ , expressed as  $\hat{x} = x + \Delta x$ :

$$\hat{x} = x + \Delta x = A^{-1}\hat{y} = A^{-1}y + A^{-1}\Delta y. \quad (1.175)$$

Using that  $\|Ax\| \leq \|A\|\|x\|$  for any norm,

$$\|\Delta x\|_2 = \|A^{-1}\Delta y\|_2 \leq \|A^{-1}\|_2 \|\Delta y\|_2. \quad (1.176)$$



## 1.6. Computational Aspects

121

To find the relative error, we divide (1.176) by the norm of  $\hat{x}$ ,

$$\frac{\|\Delta x\|_2}{\|\hat{x}\|_2} \leq \|A^{-1}\|_2 \frac{\|\Delta y\|_2}{\|\hat{x}\|_2} = \|A^{-1}\|_2 \|A\|_2 \frac{\|\Delta y\|_2}{\|A\|_2 \|\hat{x}\|_2} \leq \kappa(A) \frac{\|\Delta y\|_2}{\|\hat{y}\|_2}, \quad (1.177)$$

where  $\kappa(A)$  is called a *condition number of a matrix*  $A$ ,

$$\kappa(A) = \|A\|_2 \|A^{-1}\|_2 = \frac{\sigma_{\max}(A)}{\sigma_{\min}(A)} = \sqrt{\frac{\lambda_{\max}(A^*A)}{\lambda_{\min}(A^*A)}}. \quad (1.178a)$$

When  $A$  is a basis synthesis operator,  $\lambda_{\min}(A^*A)$  and  $\lambda_{\max}(A^*A)$  are the constants in Definition 1.34 (Riesz basis). For a Hermitian matrix (or more generally a normal matrix), the condition number is simply

$$\kappa(A) = \left| \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)} \right|. \quad (1.178b)$$

From (1.177), we see that  $\kappa(A)$  measures the sensitivity of the solution: a small amount of noise on a data vector  $y$  might grow by a factor  $\kappa(A)$  in the solution vector  $x$ . In some ill-conditioned problems, the ratio between the largest and smallest eigenvalues in (1.178) can be several orders of magnitude, sometimes leading to a useless solution. At the other extreme, the best-conditioned problems appear when  $A$  is unitary, since then  $\kappa(A) = 1$ ; the error in the solution is similar to the error in the input.

Poor conditioning can come from a large  $|\lambda_{\max}|$ , but more often, from a small  $|\lambda_{\min}|$ , that is, when the matrix  $A$  is almost singular. We would like to find out how close  $A$  is to a singular matrix. In other words, can a perturbation  $\Delta A$  lead to a singular matrix  $(A + \Delta A)$ ? It can be shown that the *minimum relative perturbation* of  $A$ , or  $\min(\|\Delta A\|_2 / \|A\|_2)$  such that  $A + \Delta A$  becomes singular, equals  $1/\kappa(A)$ . We show this in a simple case, namely, when both  $A$  and its perturbation are diagonalizable by the same unitary matrix  $U$  (this happens for certain structured matrices). Then

$$U^* A U = \Lambda \quad \text{and} \quad U^* \Delta A U = \Delta \Lambda, \quad (1.179)$$

where  $\Lambda$  is the diagonal matrix of eigenvalues of  $A$  and  $\Delta \Lambda$  is the perturbation. The minimum to perturb  $A$  into a singular matrix is

$$\min \frac{\|\Delta A\|_2}{\|A\|_2} \stackrel{(a)}{=} \min \frac{\|U^* \Delta A U\|_2}{\|U^* A U\|_2} \stackrel{(b)}{=} \frac{\min \|\Delta \Lambda\|_2}{\|\Lambda\|_2} \stackrel{(c)}{=} \left| \frac{\lambda_{\min}}{\lambda_{\max}} \right| \stackrel{(d)}{=} \frac{1}{\kappa(A)}, \quad (1.180)$$

where (a) follows from  $U$  being unitary; (b) from (1.179) and the fact that the optimization is over the perturbation, not over  $A$ ; (c) from (1.174a) and  $\Lambda - \text{diag}([0, 0, \dots, \lambda_{\min}])$  being the singular matrix closest to  $\Lambda$  and thus  $\min \|\Delta \Lambda\|_2 = |\lambda_{\min}|$ ; and (d) from (1.178b).

EXAMPLE 1.59 (CONDITIONING OF MATRICES, EXAMPLE 1.28 CONT'D) Take the matrix associated with the basis in Example 1.28(i):

$$A = [\varphi_0 \quad \varphi_1] = \begin{bmatrix} 1 & 0 \\ 0 & a \end{bmatrix}, \quad a \in (0, \infty).$$

The eigenvalues of  $A$  are 1 and  $a$ , from which the condition number follows as

$$\kappa(A) = \begin{cases} a, & a \geq 1; \\ 1/a, & a < 1. \end{cases}$$

For Example 1.28(ii),

$$A = \begin{bmatrix} \varphi_0 & \varphi_1 \end{bmatrix} = \begin{bmatrix} 1 & \cos \theta \\ 0 & \sin \theta \end{bmatrix}, \quad \theta \in (0, \pi/2]. \quad (1.181)$$

The singular value decomposition (1.213) of this matrix  $A$  leads to the condition number

$$\kappa(A) = \sqrt{\frac{1 + \cos \theta}{1 - \cos \theta}}, \quad (1.182)$$

which is plotted in Figure 1.32(a) on a log scale;  $A$  is ill conditioned as  $\theta \rightarrow 0$ , as expected.

**EXAMPLE 1.60 (AVERAGING, EXAMPLE 1.29 CONT'D)** We take another look at Example 1.29(ii) from a matrix conditioning point of view. Take the following  $N \times N$  matrix:

$$A = \begin{bmatrix} 1 & & & & \\ & \frac{1}{2} & & & \\ & & \frac{1}{3} & & \\ & & & \ddots & \\ & & & & \frac{1}{N} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \dots & 1 \end{bmatrix}, \quad (1.183)$$

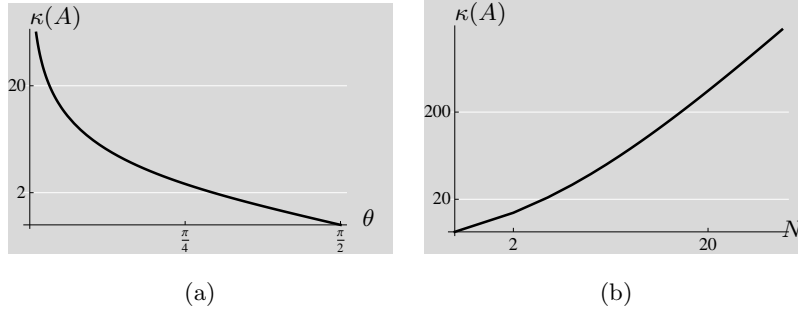
which computes the successive averages

$$x^{(k)} = \frac{1}{k} \sum_{n=0}^{k-1} x_n, \quad k = 1, 2, \dots, N.$$

While this matrix is nonsingular (a product of a diagonal matrix and a lower-triangular matrix, both with positive diagonal entries), solving  $y = Ax$  (finding the original values from the averages) is an ill-conditioned problem because the dependence of  $y$  on  $x_n$  diminishes with increasing  $n$ . Figure 1.32(b) shows the condition number  $\kappa(A)$  for  $N = 2, 3, \dots, 50$  on a log-log scale.

### 1.6.4 Solving Systems of Linear Equations

Having discussed conditioning of linear systems, let us consider algorithms to compute the solution.



**Figure 1.32:** Behaviors of the condition numbers of matrices. (a) The matrices from (1.181) with condition numbers (1.182), plotted on a log scale. (b) The matrices from (1.183), plotted on a log-log scale.

**Gaussian Elimination** The standard algorithm to solve a system of linear equations is Gaussian elimination. The algorithm uses elementary row operations to create a new system of equations  $y' = A'x$  where  $A'$  is now an upper-triangular matrix. Then using back substitution from  $x_{N-1}$  up to  $x_0$ , one finds the unknown vector  $x$ .

We can easily obtain the upper-triangular  $A'$ , by working on one column at the time and using orthogonality of length-two vectors. For example, given a vector  $[a_0 \ a_1]^T$  then  $[a_1 \ -a_0]^T$  is automatically orthogonal to it. To transform the first column, we premultiply  $A$  by the matrix  $B^{(1)}$  with entries

$$B^{(1)} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ a_{1,0} & -a_{0,0} & 0 & \cdots & 0 \\ a_{2,0} & 0 & -a_{0,0} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{N-1,0} & 0 & 0 & \cdots & -a_{0,0} \end{bmatrix}, \quad (1.184)$$

leading to a new matrix  $A^{(1)} = B^{(1)}A$  with the first column  $a_0^{(1)} = [a_{0,0} \ 0 \ \cdots \ 0]^T$ . We can continue the process by iterating on the lower right submatrix of size  $(N-1) \times (N-1)$  and so on, leading to an upper-triangular matrix

$$A^{(N-1)} = B^{(N-1)} \cdots B^{(2)} B^{(1)} A = \begin{bmatrix} \times & \times & \cdots & \times & \times \\ 0 & \times & \cdots & \times & \times \\ 0 & 0 & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \times & \times \\ 0 & 0 & \cdots & 0 & \times \end{bmatrix}. \quad (1.185)$$

The initial system of equations is thus transformed into a triangular one,

$$B^{(N-1)} \cdots B^{(2)} B^{(1)} y = A^{(N-1)} x,$$

which is solved easily by back substitution.

EXAMPLE 1.61 (TRIANGULARIZATION AND BACK SUBSTITUTION) Given a  $3 \times 3$  system  $y = Ax$  with a rank-3 matrix  $A$ , we can show that

$$B^{(1)} = \begin{bmatrix} 1 & 0 & 0 \\ a_{1,0} & -a_{0,0} & 0 \\ a_{2,0} & 0 & -a_{0,0} \end{bmatrix},$$

$$B^{(2)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & a_{2,0}a_{0,2} - a_{0,0}a_{2,1} & a_{0,0}a_{1,1} - a_{1,0}a_{0,1} \end{bmatrix}.$$

The new system is now of the form

$$y' = B^{(2)}B^{(1)}y = B^{(2)}B^{(1)}Ax = A'x, \quad (1.186)$$

with the matrix  $A'$  upper triangular and  $a'_{i,i} \neq 0$  (because  $A$  is of full rank). We can solve for  $x$  using back substitution,

$$x_2 = \frac{1}{a'_{2,2}}y'_2,$$

$$x_1 = \frac{1}{a'_{1,1}}(y'_1 - a'_{1,2}x_2),$$

$$x_0 = \frac{1}{a'_{0,0}}(y'_0 - a'_{0,1}x_1 - a'_{0,2}x_2).$$

The form of the solution indicates that conditioning is a key issue, since if some  $a'_{i,i}$  is close to zero, the solution might be ill behaved.

In the above discussion, we did not discuss ordering of operations, or choosing a particular row to zero the entries of a particular column. In practice, this choice, called choosing the *pivot*, is important for the numerical behavior of the algorithm. In general, the solution of a linear system depends on whether the vector  $y$  belongs to the range (column space) of  $A$ . If not, there is no possible solution. If it does, and the columns are linearly independent, the solution is unique; otherwise there is an infinite number of solutions.

**Cost of Gaussian Elimination** The cost of Gaussian elimination is dominated by the cost of forming the product  $B^{(N-1)} \dots B^{(2)}B^{(1)}A$ , resulting in the upper triangular matrix  $A^{(N-1)}$ . Multiplying  $B^{(1)}$  and  $A$  uses  $\Theta(N^2)$  multiplications and additions, so after  $N - 2$  additional such multiplications  $A^{(N-1)}$  is formed with  $\Theta(N^3)$  multiplications and additions. The other steps are cheaper: forming  $B^{(N-1)} \dots B^{(2)}B^{(1)}y$  has  $\Theta(N^2)$  cost; and back substitution requires  $N$  divisions and  $\Theta(N^2)$  multiplications and additions. A careful accounting gives a total multiplicative cost of about  $\frac{1}{3}N^3$ .

One can use Gaussian elimination to calculate the inverse of a matrix  $A$ . Solving  $Ax = e_k$ , where  $e_k$  is the  $k$ th vector of the standard basis, gives the  $k$ th column of  $A^{-1}$ . The cost of finding each column in this manner is  $\Theta(N^3)$ , so this overall algorithm has  $\Theta(N^4)$  cost for the full inverse. This is very inefficient. An

inversion algorithm that forms and saves the *LU decomposition* of  $A$  while solving  $Ax = e_0$  with Gaussian elimination has cost approximately  $\frac{4}{3}N^3$ . Note that we rarely calculate  $A^{-1}$  explicitly; solving  $y = Ax$  is no cheaper with  $A^{-1}$  than with the LU decomposition of  $A$ .

**Sparse Matrices and Iterative Solutions of Systems of Linear Equations** If the matrix–vector product  $Ax$  is easy to compute, that is, with a cost substantially smaller than  $N^2$ , then iterative solvers can be considered. This is the case when  $A$  is sparse or banded as in (1.229) and has only  $O(N)$  nonzero entries.

An iterative algorithm computes a new approximate solution from an old approximate solution with an update step; if properly designed, it will converge to the solution. The basic idea is to write  $A = D - B$ , transforming  $y = Ax$  into

$$Dx = Bx + y. \quad (1.187)$$

The update step is

$$x^{(k+1)} = D^{-1}(Bx^{(k)} + y), \quad (1.188)$$

which has the desired solution as a fixed point. If  $D^{-1}$  is easy to compute (for example, it is diagonal), then this is a valid approach.

To study the error, let  $e^{(k)} = x - x^{(k)}$ . Subtracting

$$Dx^{(k+1)} = Bx^{(k)} + y$$

from (1.187) yields

$$e^{(k+1)} = D^{-1}Be^{(k)} = (D^{-1}B)^{k+1}e^{(0)},$$

with  $e^{(0)}$  the initial error. The algorithm will converge to the true solution for any initial guess  $x^{(0)}$  if and only if  $(D^{-1}B)^{k+1} \rightarrow 0$  as  $k \rightarrow \infty$ , which happens when all the eigenvalues of  $D^{-1}B$  are smaller than 1 in absolute value.

**EXAMPLE 1.62 (ITERATIVE SOLUTION OF A TOEPLITZ SYSTEM)** Take a Toeplitz matrix  $A$  as in (1.228) and write it as  $D - B = I - (I - A)$ . Then (1.188) reduces to

$$x^{(k+1)} = (I - A)x^{(k)} + y. \quad (1.189)$$

Note that  $B = I - A$  is still Toeplitz, allowing a fast multiplication for evaluating  $Bx^{(k)}$ , as will be seen in Section 2.9. If the eigenvalues of  $D^{-1}B = I - A$  are smaller than 1 in absolute value, the iterative algorithm will converge.

As an example, consider the matrix describing a two-point sum,

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix},$$

in the system  $Ax = y$  with  $y = [1 \ 3 \ 5 \ 7]^T$ . The eigenvalues of  $(I - A)$  are all 0, and thus, the algorithm will converge. For example, start with an all-zero

vector  $x^{(0)}$ . The iterative procedure (1.189) produces the following:

$$x^{(1)} = \begin{bmatrix} 1 \\ 3 \\ 5 \\ 7 \end{bmatrix}, \quad x^{(2)} = \begin{bmatrix} 1 \\ 2 \\ 2 \\ 2 \end{bmatrix}, \quad x^{(3)} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 5 \end{bmatrix}, \quad x^{(4)} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix},$$

and converges in the fourth step ( $x^{(n)} = x^{(4)}$  for  $n \geq 5$ ).

Among iterative solvers of large systems of linear equations, Kaczmarz's algorithm has an intuitive geometric interpretation.

**EXAMPLE 1.63 (KACZMARZ'S ALGORITHM)** Consider a square system of linear equations  $y = Ax$  with  $A$  real and of full rank. We can look for the solution  $x = [x_0 \ x_1 \ \dots \ x_{N-1}]^T$  in two ways, concentrating on either the columns or rows of  $A$ . When concentrating on the columns  $\{v_0, v_1, \dots, v_{N-1}\}$ , we see the solution  $x$  as giving the coefficients to form  $y$  as a linear combination of columns:

$$\sum_{n=0}^{N-1} x_n v_n = y. \quad (1.190a)$$

When concentrating on the rows  $\{r_0^T, r_1^T, \dots, r_{N-1}^T\}$ , we see the solution  $x$  as the vector that has all the correct inner products:

$$\langle x, r_n \rangle = y_n, \quad n = 0, 1, \dots, N-1. \quad (1.190b)$$

Kaczmarz's algorithm uses the row-based view. Normalize  $r_n$  to unit norm,  $\gamma_n = r_n / \|r_n\|$ . Then (1.190b) becomes

$$\langle x, \gamma_n \rangle = \frac{y_n}{\|r_n\|} = y'_n, \quad n = 0, 1, \dots, N-1. \quad (1.191)$$

The idea of the Kaczmarz's algorithm is to iteratively satisfy the constraints (1.191). Starting with an initial guess  $x^{(-1)}$ , the first update step is  $N$  computations

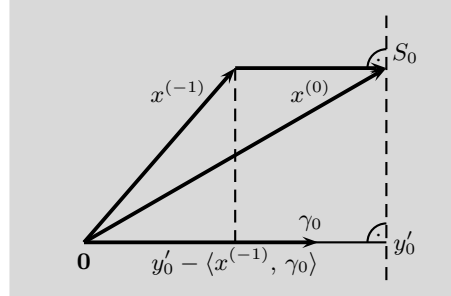
$$x^{(n)} = x^{(n-1)} + (y'_n - \langle x^{(n-1)}, \gamma_n \rangle) \gamma_n, \quad n = 0, 1, \dots, N-1, \quad (1.192)$$

called a sweep. With the update (1.192),  $x^{(n)}$  satisfies

$$\langle x^{(n)}, \gamma_n \rangle = \langle x^{(n-1)}, \gamma_n \rangle + y'_n \underbrace{\langle \gamma_n, \gamma_n \rangle}_{=1} - \underbrace{\langle \gamma_n, \gamma_n \rangle}_{=1} \langle x^{(n-1)}, \gamma_n \rangle = y'_n,$$

as desired. At the end of this sweep,  $x^{(N-1)}$  will most likely not satisfy  $\langle x^{(N-1)}, \gamma_0 \rangle = y'_0$ , and thus, further sweeps are required.

To understand the algorithm geometrically, note that the update  $x^{(0)}$  is the orthogonal projection of the initial guess  $x^{(-1)}$  onto the affine subspace  $S_0$  orthogonal to the subspace spanned by  $\gamma_0$  and at the distance  $y'_0$  from the origin



**Figure 1.33:** One step of Kaczmarz's algorithm. The update  $x^{(0)}$  is the orthogonal projection of the initial guess  $x^{(-1)}$  onto the affine subspace  $S_0$  orthogonal to the subspace spanned by  $\gamma_0$  and at the distance  $y'_0$  from the origin  $\mathbf{0}$ .

as in Figure 1.33. The desired solution is  $x = \cap_{i=1}^n S_i$ . Convergence is geometric in the number of sweeps, with a constant depending on how close to orthogonal the vectors  $\gamma_n$  are. When the rows of  $A$  are orthogonal, convergence is in one sweep (see Exercise 1.61).

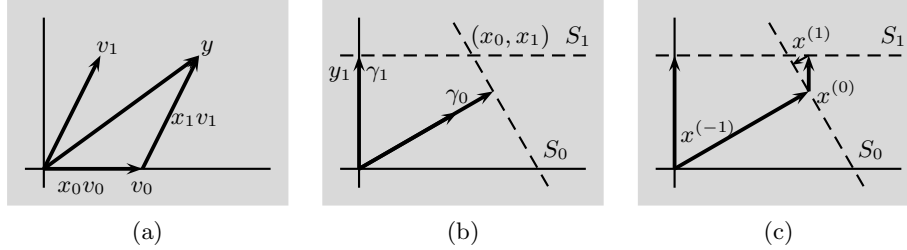
In Figure 1.34, we show three different interpretations of solving the system of linear equations

$$\begin{bmatrix} \frac{\sqrt{3}}{2} & \frac{1}{2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{3}+1}{2} \\ 1 \end{bmatrix}, \quad (1.193)$$

which has the solution  $x = [1 \ 1]^T$ . The rows are already of norm 1, and thus  $\gamma_n = r_n$  and  $y'_n = y_n$ . Figure 1.34(a) shows the solution as a linear combination (1.190a) of column vectors  $\{v_0, v_1\}$ . In this particular case, it turns out that  $y = v_0 + v_1$  exactly since  $x_0 = x_1 = 1$ . Figure 1.34(b) shows the solution as the intersection of linear constraints (1.190b), that is, the intersection of the two subspaces  $S_0$  and  $S_1$ , orthogonal to the spans of  $\gamma_0$  and  $\gamma_1$ , and at the distance from the origin  $y_0$  and  $y_1$ , respectively. The intersection of  $S_0$  and  $S_1$  is exactly the solution  $x = [1 \ 1]^T$ . Finally, Figure 1.34(c) shows a few steps of the iterative algorithm (1.192), starting with  $x^{(-1)} = 0$ .

**Complexity of Solving a Linear System of Equations** The cost of the Gaussian elimination algorithm (or any other algorithm) provides an upper bound to the complexity of solving a general linear system of equations. The precise multiplicative complexity has not been proven.

If the matrix  $A$  is structured, computational savings can be achieved. For example, we will see in the next chapter that when the matrix is circulant as in (1.227), the cost is  $O(N \log_2 N)$  as in (2.261), because the discrete Fourier transform (DFT) diagonalizes the circulant convolution operator, and many fast algorithms for computing the DFT exist. Also, solvers with  $O(N^2)$  cost exist for the cases where the matrix is Toeplitz as in (1.228) or Vandermonde as in (1.230). *Further Reading* gives pointers to literature on these algorithms.



**Figure 1.34:** Different views of solving the system of linear equations in (1.193). (a) As a linear combination (1.190a) of column vectors  $\{v_0, v_1\}$ . (b) As the intersection of linear constraints (1.190b). (c) As the solution to the iterative algorithm (1.192), starting with  $x^{(-1)} = 0$ .

## Appendix

### 1.A Elements of Analysis and Topology

This appendix reviews some basic elements of real analysis (under Lebesgue measure as applicable) and the standard topology on the real line. Some material is adapted from [100, 122].

#### 1.A.1 Basic Definitions

**Sets** Let  $W$  be a subset of  $\mathbb{R}$ . An *upper bound* is a number  $M$  such that every  $w$  in  $W$  satisfies  $w \leq M$ . The smallest of all upper bounds is called the *supremum* of  $W$  and denoted  $\sup W$ ; if no upper bound exists,  $\sup W = \infty$ . A *lower bound* is a number  $m$  such that every  $w$  in  $W$  satisfies  $w \geq m$ . The largest of all lower bounds is called the *infimum* of  $W$  and denoted  $\inf W$ ; if no lower bound exists,  $\inf W = -\infty$ .

The *essential supremum* and *essential infimum* are defined similarly but are based on bounds that can be violated by a countable number of points. An *essential upper bound* is a number  $M$  such that at most a countable number of  $w$  in  $W$  violates  $w \leq M$ . The smallest of all essential upper bounds is called the *essential supremum* of  $W$  and denoted  $\text{ess sup } W$ ; if no essential upper bound exists,  $\text{ess sup } W = \infty$ . An *essential lower bound* is a number  $m$  such that at most a countable number of  $w$  in  $W$  violates  $w \geq m$ . The smallest of all essential lower bounds is called the *essential infimum* of  $W$  and denoted  $\text{ess inf } W$ ; if no essential lower bound exists,  $\text{ess inf } W = -\infty$ .

**Topology** Let  $W$  be a subset of  $\mathbb{R}$ . An element  $w \in W$  is an *interior point* if there is an  $\varepsilon > 0$  such that  $(w - \varepsilon, w + \varepsilon) \subset W$ . A set is *open* if all its points are interior points. Facts about open sets include the following:

- (i)  $\mathbb{R}$  is open.
- (ii)  $\emptyset$  is open.



- (iii) The union of *any* collection of open sets is open.
- (iv) The intersection of *finitely many* open sets is open.

A set is *closed* when its complement is open. By complementing the sets in the list above, facts about closed sets include the following:

- (i)  $\emptyset$  is closed.
- (ii)  $\mathbb{R}$  is closed.
- (iii) The intersection of *any* collection of closed sets is closed.
- (iv) The union of *finitely many* closed sets is closed.

The *closure* of a set  $W$ , denoted by  $\overline{W}$ , is the intersection of all closed sets containing  $W$ . It is also the set of all limit points of convergent sequences in the set. A set is closed if and only if it is equal its closure. Also, a set is closed if and only if it contains the limit of every convergent sequence in the set.

**Functions** A *function*  $x$  takes an *argument* (input)  $t$  and produces a *value* (output)  $x(t)$ . The acceptable values of the argument form the *domain*, while the possible function values form the *range*, also called the *image*. If the range is a subset of a larger set, that set is termed the *codomain*. The notation  $x : D \rightarrow C$  indicates that  $x$  is a function with domain  $D$  and codomain  $C$ . A *composition* of functions uses the output of one function as the input to another. For example,  $y(t) = x_2(x_1(t))$  will be denoted as  $y : D \xrightarrow{x_1} C_1 \xrightarrow{x_2} C_2$ . A function that maps a vector space into a vector space is called an *operator*.

A function is *injective* if  $x(t_1) = x(t_2)$  implies that  $t_1 = t_2$ . In other words, different values of the function must have been produced by different arguments. A function is *surjective* if the range equals the codomain, that is, if for every  $y \in C$ , there exists a  $t \in D$ , such that  $x(t) = y$ . A function is *bijective* if it is both injective and surjective. A bijective function  $x : D \rightarrow C$  has an *inverse*  $x^{-1} : C \rightarrow D$  such that  $x^{-1}(x(t)) = t$  for all  $t \in D$  and  $x(x^{-1}(y)) = y$  for all  $y \in C$ . These concepts are illustrated in Figure 1.35.

## 1.A.2 Convergence

**Sequences** A sequence of numbers  $a_0, a_1, \dots$  is said to *converge* to the number  $a$  (written  $\lim_{k \rightarrow \infty} a_k = a$ ) when the following holds:

for any  $\varepsilon > 0$  there exists a number  $K_\varepsilon$  such that  $|a_k - a| < \varepsilon$  for every  $k > K_\varepsilon$ .

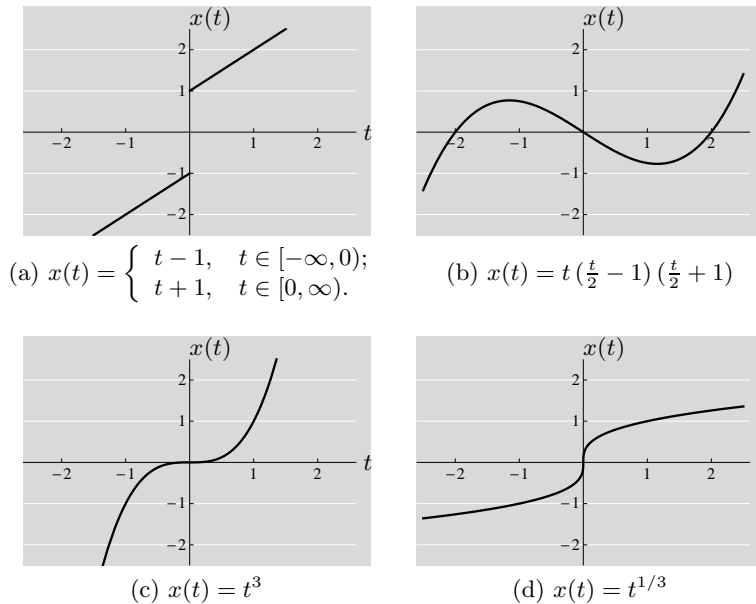
The sequence is said to *diverge* if it does not converge to any (finite) number. It *diverges to  $\infty$*  (written  $\lim_{k \rightarrow \infty} a_k = \infty$ ) when the following holds:

for any  $M$  there exists a number  $K_M$  such that  $a_k > M$  for every  $k > K_M$ .

Similarly, it *diverges to  $-\infty$*  (written  $\lim_{k \rightarrow \infty} a_k = -\infty$ ) when the following holds:

for any  $M$  there exists a number  $K_M$  such that  $a_k < M$  for every  $k > K_M$ .

A few properties of convergence of sequences are derived in Exercise 1.63.



**Figure 1.35:** Examples of different types of functions  $x : \mathbb{R} \rightarrow \mathbb{R}$ . (a) Injective, but not surjective; the range is  $\mathcal{R} = (\infty, -1] \cup [1, \infty)$ . (b) Surjective, but not injective. (c) Bijective (both injective and surjective). (d) Inverse of the bijective function from (c).

**Series** Let  $a_0, a_1, \dots$  be numbers. The numbers  $s_n = \sum_{k=0}^n a_k$ ,  $n = 0, 1, \dots$ , are called *partial sums* of the (*infinite*) *series*  $\sum_{k=0}^{\infty} a_k$ . The series is said to *converge* when the sequence of partial sums converges. We write  $\sum_{k=0}^{\infty} a_k = \infty$  when the partial sums diverge to  $\infty$  and  $\sum_{k=0}^{\infty} a_k = -\infty$  when the partial sums diverge to  $-\infty$ .

The series  $\sum_{k=0}^{\infty} a_k$  is said to *converge absolutely* when  $\sum_{k=0}^{\infty} |a_k|$  converges. A series that converges but does not converge absolutely is said to *converge conditionally*. The definition of convergence takes the terms of a series in a particular order. When a series is absolutely convergent, its terms can be reordered without altering its convergence or its value; otherwise not.<sup>25</sup> The *doubly-infinite series*  $\sum_{k=-\infty}^{\infty} a_k$  does not have a single natural choice of partial sums, so it is said to converge when it converges absolutely.

Tests for convergence of series are reviewed in Exercise 1.64 and a few useful series are explored in Exercise 1.65.

**Functions** A sequence of real-valued functions  $x_0, x_1, \dots$  *converges pointwise* when, for any fixed  $t$ , the sequence of numbers  $x_0(t), x_1(t), \dots$  converges. More explicitly, suppose the functions have a common domain  $D$ . They converge pointwise to func-

<sup>25</sup>A strange and wonderful fact known as the Riemann series theorem is that a conditionally convergent series can be rearranged to converge to any desired value or to diverge!

tion  $x$  with domain  $D$  when, for any  $\varepsilon > 0$  and  $t \in D$ , there exists a number  $K_{\varepsilon,t}$  (depending on  $\varepsilon$  and  $t$ ) such that

$$|x_k(t) - x(t)| < \varepsilon \quad \text{for all } k > K_{\varepsilon,t}.$$

A more restrictive form of convergence does not allow  $K_{\varepsilon,t}$  to depend on  $t$ . A sequence  $x_0, x_1, \dots$  of real-valued functions on some domain  $D$  *converges uniformly* to  $x : D \rightarrow \mathbb{R}$  when, for any  $\varepsilon > 0$ , there exists a number  $K_\varepsilon$  (depending on  $\varepsilon$ ) such that

$$|x_k(t) - x(t)| < \varepsilon \quad \text{for all } t \in D \text{ and all } k > K_\varepsilon.$$

Uniform convergence implies pointwise convergence. Furthermore, if a sequence of continuous functions is uniformly convergent, the limit function is necessarily continuous.

### 1.A.3 Interchange Theorems

Many derivations in analysis involve interchanging the order of sums, integrals, and limits without changing the result. Without appropriate caution, this may be simply incorrect; refer again to Footnote 25.

Two nested summations can be seen as a single sum over a two-dimensional index set. Thus, since absolute convergence allows rearrangement of the terms in a sum, it allows changing the order of summations:

$$\sum_{n=0}^{\infty} \sum_{k=0}^{\infty} |x_{n,k}| < \infty \quad \text{implies} \quad \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} x_{n,k} = \sum_{k=0}^{\infty} \sum_{n=0}^{\infty} x_{n,k}. \quad (1.194)$$

This extends to doubly-infinite summations and more than two summations.

The analogous result for integrals is called *Fubini's theorem*:

$$\begin{aligned} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |x(t_1, t_2)| dt_1 dt_2 < \infty \quad \text{implies} \\ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(t_1, t_2) dt_1 dt_2 = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(t_1, t_2) dt_2 dt_1. \end{aligned} \quad (1.195)$$

This extends to more than two integrals.

Interchange of summation and integration can be justified by uniform convergence. Suppose a sequence of partial sums  $s_n(t) = \sum_{k=0}^n x_k(t)$ ,  $n = 0, 1, \dots$ , is uniformly convergent to  $s(t)$  on  $[a, b]$ . Then the series may be integrated term by term:

$$\int_a^b \sum_{k=0}^{\infty} x_k(t) dt = \sum_{k=0}^{\infty} \int_a^b x_k(t) dt. \quad (1.196)$$

This result extends to infinite intervals as well.

Uniform convergence is rather restrictive as a justification for (1.196) since changing any of the  $x_k$ s on a set of zero measure would not change either side of the equality. Another result on interchanging summation and integration is as follows.

If  $|x_k(t)| \leq y_k(t)$  for all  $k \in \mathbb{N}$  and almost all  $t \in [a, b]$  and  $\sum_{k=0}^{\infty} y_k(t)$  converges for almost all  $t \in [a, b]$  and  $\sum_{k=0}^{\infty} \int_a^b y_k(t) dt < \infty$ , then (1.196) holds. At the heart of this result is an application of the dominated convergence theorem to the sequence of partial sums  $s_n(t) = \sum_{k=0}^n x_k(t)$ ,  $n = 0, 1, \dots$

The *dominated convergence theorem* enables interchange of a limit with an integral: Let  $x_0, x_1, \dots$  be real-valued functions such that  $\lim_{k \rightarrow \infty} x_k(t) = x(t)$  for almost all  $t \in \mathbb{R}$ . If there exists a (nonnegative) real-valued function  $y$  such that for all  $k \in \mathbb{N}$ ,

$$|x_k(t)| \leq y(t) \quad \text{holds for almost all } t \in \mathbb{R} \quad \text{and} \quad \int_{-\infty}^{\infty} y(t) dt < \infty,$$

then  $x$  is integrable, and

$$\int_{-\infty}^{\infty} \left( \lim_{k \rightarrow \infty} x_k(t) \right) dt = \lim_{k \rightarrow \infty} \int_{-\infty}^{\infty} x_k(t) dt. \quad (1.197)$$

### 1.A.4 Inequalities

See [149] for elementary proofs and further details.

**Minkowski's Inequality** For any  $p \in [1, \infty)$ ,

$$\left( \sum_{k \in \mathbb{Z}} |x_k + y_k|^p \right)^{1/p} \leq \left( \sum_{k \in \mathbb{Z}} |x_k|^p \right)^{1/p} + \left( \sum_{k \in \mathbb{Z}} |y_k|^p \right)^{1/p}. \quad (1.198a)$$

This establishes that the  $\ell^p$  norm (1.36a) satisfies the triangle inequality in Definition 1.9. Also, for any  $p \in [1, \infty)$ ,

$$\left( \int_a^b |x(t) + y(t)|^p dt \right)^{1/p} \leq \left( \int_a^b |x(t)|^p dt \right)^{1/p} + \left( \int_a^b |y(t)|^p dt \right)^{1/p}, \quad (1.198b)$$

establishing that the  $\mathcal{L}^p$  norm (1.38a) satisfies the triangle inequality.

Analogues of (1.198) hold for  $\ell^\infty$  and  $\mathcal{L}^\infty$  as well:

$$\sup_{k \in \mathbb{Z}} |x_k + y_k| \leq \sup_{k \in \mathbb{Z}} |x_k| + \sup_{k \in \mathbb{Z}} |y_k| \quad (1.199a)$$

and

$$\sup_{t \in \mathbb{R}} |x(t) + y(t)| \leq \sup_{t \in \mathbb{R}} |x(t)| + \sup_{t \in \mathbb{R}} |y(t)|. \quad (1.199b)$$

**Hölder's Inequality** Let  $p$  and  $q$  in  $[1, \infty]$  satisfy  $1/p + 1/q = 1$  with the convention that  $1/\infty = 0$  is allowed. Then  $p$  and  $q$  are called *Hölder conjugates* and

$$\|xy\|_1 \leq \|x\|_p \|y\|_q \quad (1.200)$$

## 1.B. Elements of Linear Algebra

133

for sequences or functions  $x$  and  $y$ , with equality if and only if  $|x|^p$  and  $|y|^q$  are scalar multiples of each other. The case of  $p = q = 2$  is the Cauchy–Schwarz inequality (1.24).

Specializing (1.200) to sequences gives

$$\sum_{k \in \mathbb{Z}} |x_k y_k| \leq \left( \sum_{k \in \mathbb{Z}} |x_k|^p \right)^{1/p} \left( \sum_{k \in \mathbb{Z}} |y_k|^q \right)^{1/q} \quad (1.201a)$$

for finite  $p$  and  $q$ , and

$$\sum_{k \in \mathbb{Z}} |x_k y_k| \leq \left( \sup_{k \in \mathbb{Z}} |x_k| \right) \left( \sum_{k \in \mathbb{Z}} |y_k| \right) \quad (1.201b)$$

for  $p = \infty$ . Similarly, for functions

$$\int_{-\infty}^{\infty} |x(t) y(t)| dt \leq \left( \int_{-\infty}^{\infty} |x(t)|^p dt \right)^{1/p} \left( \int_{-\infty}^{\infty} |y(t)|^q dt \right)^{1/q} \quad (1.202a)$$

for finite  $p$  and  $q$ , and

$$\int_{-\infty}^{\infty} |x(t) y(t)| dt \leq \left( \sup_{t \in \mathbb{R}} |x(t)| \right) \left( \int_{-\infty}^{\infty} |y(t)| dt \right) \quad (1.202b)$$

for  $p = \infty$ .

## 1.A.5 Integration by Parts

Integration by parts transforms an integral into another integral, possibly easier to solve. It can be written very compactly as

$$\int u dv = uv - \int v du \quad (1.203a)$$

or more explicitly as

$$\int_a^b u(t) v'(t) dt = u(t) v(t) \Big|_{t=a}^{t=b} - \int_a^b v(t) u'(t) dt. \quad (1.203b)$$

## 1.B Elements of Linear Algebra

This appendix reviews basic concepts in linear algebra. Good sources for more details include [76, 140]. Contrary to the standard convention in finite-dimensional linear algebra, we start all indexing at 0 rather than 1; this facilitates consistency throughout the book.

### 1.B.1 Basic Definitions and Properties

We say a matrix  $A$  is  $M \times N$  or in  $\mathbb{C}^{M \times N}$  when it has  $M$  rows and  $N$  columns. It is a linear operator mapping  $\mathbb{C}^N$  into  $\mathbb{C}^M$ . When  $M = N$ , the matrix is called *square*; otherwise it is *rectangular*.<sup>26</sup> An  $M \times 1$  matrix is called a *column vector*, a  $1 \times N$  matrix a *row vector*, and a  $1 \times 1$  matrix a *scalar*. Unless stated explicitly otherwise,  $A_{m,n}$  denotes the row- $m$ , column- $n$  entry of matrix  $A$ .

**Basic Operations** Addition of matrices is element-by-element, so matrices can only be added if they have the same dimensions. The product of  $A \in \mathbb{C}^{M \times P}$  and  $B \in \mathbb{C}^{P \times N}$  is defined only when  $P = Q$ , in which case it is given by

$$(AB)_{m,n} = \sum_{k=0}^{P-1} A_{m,k} B_{k,n}, \quad \begin{array}{l} m = 0, 1, \dots, M-1, \\ n = 0, 1, \dots, N-1. \end{array} \quad (1.204)$$

Entries  $A_{m,n}$  are on the (*main*) *diagonal* if  $m = n$ . A square matrix with unit diagonal entries and zero off-diagonal entries is called an *identity matrix* and denoted by  $I$ . It is the identity element under matrix multiplication. For square matrices  $A$  and  $B$ , if  $AB = I$  and  $BA = I$ ,  $B$  is called the *inverse* of  $A$  and is written as  $A^{-1}$ . If no such  $B$  exists,  $A$  is called *singular*. When the inverses exist,  $(AB)^{-1} = B^{-1} A^{-1}$ . Rectangular matrices do not have inverses. Instead, a short matrix can have a *right* inverse  $B$ , so  $AB = I$ ; similarly, a tall matrix can have a *left* inverse  $B$ , so  $BA = I$ .

If  $A_{m,n} = B_{n,m}$  for all  $m$  and  $n$ , we write  $A = B^T$ ; we call  $B$  the *transpose* of  $A$ . If  $A_{m,n} = B_{n,m}^*$  for all  $m$  and  $n$ , we write  $A = B^*$ ; we call  $B$  the *Hermitian transpose* of  $A$ . Here,  $*$  denotes both complex conjugation of a scalar and the combination of complex conjugation and transposition of a matrix. In general,  $(AB)^T = B^T A^T$  and  $(AB)^* = B^* A^*$ .

**Determinant** The *determinant* maps a square matrix into a scalar value. It is defined recursively, with  $\det(a) = a$  for a scalar  $a$  and

$$\det(A) = \sum_{k=0}^{N-1} (-1)^{i+k} \det(M_{i,k}) A_{i,k} = \sum_{k=0}^{N-1} C_{i,k} A_{i,k}$$

for  $A \in \mathbb{C}^{N \times N}$ , where *minor*  $M_{i,k}$  is the  $(N-1) \times (N-1)$  matrix obtained by deleting the  $i$ th row and  $k$ th column of  $A$ ; *cofactor*  $C_{i,k} = (-1)^{i+k} \det(M_{i,k})$  will be used later to define the adjugate. This definition is valid because the same result is obtained for any choice of  $i \in \{1, 2, \dots, N\}$ ; to simplify computations, one may choose  $i$  to minimize the number of nonzero terms in the sum.

The determinant of  $A \in \mathbb{C}^{N \times N}$  has several useful properties, including:

- (i) For a scalar  $\alpha$ ,  $\det(\alpha A) = \alpha^N \det(A)$ .
- (ii) If  $B$  is obtained by interchanging two rows or two columns of  $A$ , then  $\det(B) = -\det(A)$ .

<sup>26</sup>We sometimes call a matrix with  $M > N$  *tall*; similarly, we call a matrix with  $M < N$  *short*.

- (iii)  $\det(A^T) = \det(A)$ .
- (iv)  $\det(AB) = \det(A)\det(B)$ .
- (v) If  $A$  is triangular, that is, all of its elements above or below the main diagonal are 0,  $\det(A)$  is the product of the diagonal elements of  $A$ .
- (vi)  $A$  is singular if and only if  $\det(A) = 0$ .

The final property relating the determinant to invertibility has both a geometric interpretation and a connection to a formula for a matrix inverse:

- (i) When the matrix is real, the determinant is the volume of the parallelepiped that has the column vectors of the matrix as edges. Thus, a zero determinant indicates linear dependence of the columns of the matrix, since the parallelepiped is not of full dimension. (The row vectors lead to a different parallelepiped with the same volume.)
- (ii) The inverse of a nonsingular matrix  $A$  is given by Cramer's formula:

$$A^{-1} = \frac{\text{adj}(A)}{\det(A)}, \quad (1.205)$$

where the *adjugate* of  $A$  is the transpose of the matrix of cofactors of  $A$ :  $(\text{adj}(A))_{i,k} = C_{k,i}$ . Cramer's formula is useful for finding inverses of small matrices by hand and as an analytical tool; it does not yield computationally-efficient techniques for inversion.

**Range, Null Space, and Rank** Associated with any matrix  $A \in \mathbb{R}^{M \times N}$  are four fundamental subspaces. The *range* or *column space* of  $A$  is the span of the columns of  $A$  and thus a subspace of  $\mathbb{R}^M$ ; it can be written as

$$\mathcal{R}(A) = \text{span}(\{a_0, a_1, \dots, a_{N-1}\}) = \{y \in \mathbb{R}^M \mid y = Ax \text{ for some } x \in \mathbb{R}^N\}, \quad (1.206a)$$

where  $a_0, a_1, \dots, a_{N-1}$  are the columns of  $A$ . Linear combinations of rows of  $A$  are all row vectors  $y^T A$  where  $y \in \mathbb{R}^M$ . Taking these as column vectors gives the *row space* of  $A$ , which is the range of  $A^T$  and a subspace of  $\mathbb{R}^N$ :

$$\mathcal{R}(A^T) = \text{span}(\{b_0^T, b_1^T, \dots, b_{M-1}^T\}) = \{x \in \mathbb{R}^N \mid x = A^T y \text{ for some } y \in \mathbb{R}^M\}, \quad (1.206b)$$

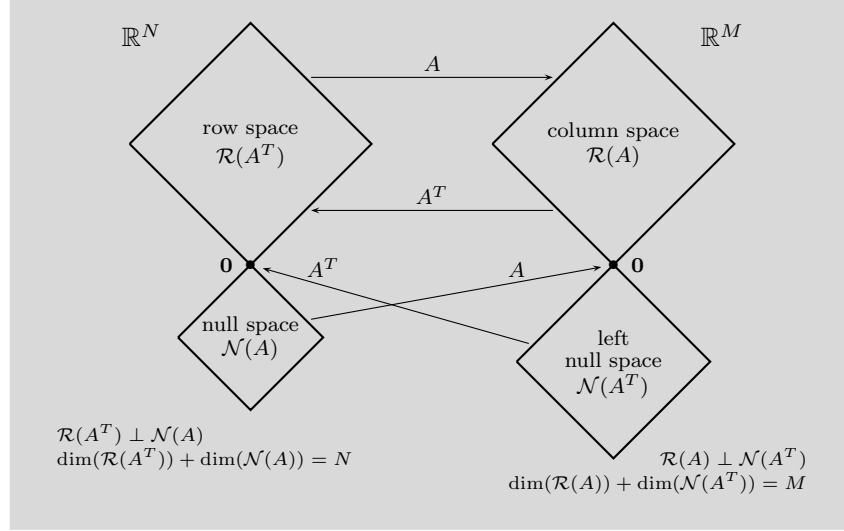
where  $b_0, b_1, \dots, b_{M-1}$  are the rows of  $A$ . The *null space* or *kernel* of  $A$  is the set of vectors that  $A$  maps to  $\mathbf{0}$  (a subspace of  $\mathbb{R}^N$ ):

$$\mathcal{N}(A) = \{x \in \mathbb{R}^N \mid Ax = \mathbf{0}\}. \quad (1.206c)$$

The *left null space* is the set of vectors mapped to zero when multiplied on the right by  $A$ , taken as column vectors. Since  $y^T A = \mathbf{0}$  is equivalent to  $A^T y = \mathbf{0}$ , the left null space of  $A$  is the null space of  $A^T$  (a subspace of  $\mathbb{R}^M$ ):

$$\mathcal{N}(A^T) = \{y \in \mathbb{R}^M \mid A^T y = \mathbf{0}\}. \quad (1.206d)$$

The four fundamental subspaces provide orthogonal decompositions of  $\mathbb{R}^N$  and  $\mathbb{R}^M$  as depicted in Figure 1.36. As shown, the null space is the orthogonal



**Figure 1.36:** The four fundamental subspaces associated with a real matrix  $A \in \mathbb{R}^{M \times N}$ . The matrix determines an orthogonal decomposition of  $\mathbb{R}^N$  into the row space of  $A$  and the null space of  $A$ ; and an orthogonal decomposition of  $\mathbb{R}^M$  into the column space (range) of  $A$  and the left null space of  $A$ . The column and row spaces of  $A$  have the same dimension, which equals the rank of  $A$ . (Figure inspired by the cover of [141].)

Space	Symbol	Definition	Dimension
Column space (range)	$\mathcal{R}(A)$	$\{y \in \mathbb{C}^M \mid y = Ax \text{ for some } x \in \mathbb{C}^N\}$	$\text{rank}(A)$
Left null space	$\mathcal{N}(A^*)$	$\{y \in \mathbb{C}^M \mid A^*y = 0\}$	$M - \text{rank}(A)$
		$\dim(\mathcal{R}(A)) + \dim(\mathcal{N}(A^*)) = M$	
Row space	$\mathcal{R}(A^*)$	$\{x \in \mathbb{C}^N \mid x = A^*y \text{ for some } y \in \mathbb{C}^M\}$	$\text{rank}(A)$
Null space (kernel)	$\mathcal{N}(A)$	$\{x \in \mathbb{C}^N \mid Ax = 0\}$	$N - \text{rank}(A)$
		$\dim(\mathcal{R}(A^*)) + \dim(\mathcal{N}(A)) = N$	

**Table 1.2:** Summary of spaces and related characteristics for a complex matrix  $A \in \mathbb{C}^{M \times N}$  (illustrated in Figure 1.36 for a real matrix  $A \in \mathbb{R}^{M \times N}$ ).

complement of the row space; the left null space is the orthogonal complement of the range (column space);  $A$  maps the null space to  $\mathbf{0}$ ;  $A^T$  maps the left null space to  $\mathbf{0}$ ; and  $A$  and  $A^T$  map between the row space and column space, which are of equal dimension. Properties of the subspaces are summarized for the complex case in Table 1.2.

The *rank* is defined by

$$\text{rank}(A) = \dim(\mathcal{R}(A)).$$



It satisfies  $\text{rank}(A) = \text{rank}(A^*)$  and  $\text{rank}(AB) \leq \min(\text{rank}(A), \text{rank}(B))$ .

**Systems of Linear Equations and Least Squares** The product  $Ax$  describes a linear combination of the columns of  $A$  weighted by the entries of  $x$ . In solving a system of linear equations,

$$Ax = y, \quad \text{where } A \in \mathbb{R}^{M \times N}, \quad (1.207)$$

we encounter the following possibilities depending on whether  $y$  belongs to the range (column space) of  $A$ ,  $y \in \mathcal{R}(A)$ , and whether the columns of  $A$  are linearly independent:

- (i) *Unique solution*: If  $y$  belongs to the range of  $A$  and the columns of  $A$  are linearly independent ( $\text{rank}(A) = N$ ), there is a unique solution.
- (ii) *Infinitely many solutions*: If  $y$  belongs to the range of  $A$  and the columns of  $A$  are not linearly independent ( $\text{rank}(A) < N$ ), there are infinitely many solutions.
- (iii) *No solution*: If  $y$  does not belong to the range of  $A$ , there is no solution. Only approximations are possible.

Cases with and without solutions are unified by looking for a *least squares* solution  $\hat{x}$ , meaning one that minimizes  $\|y - \hat{y}\|_2$ , where  $\hat{y} = A\hat{x}$ . This is obtained from the orthogonality principle: the error  $y - \hat{y}$  is orthogonal to the range of  $A$ , leading to the *normal equations*,

$$A^T A \hat{x} = A^T y. \quad (1.208a)$$

When  $A^T A$  is invertible ( $\text{rank}(A) = N$ ), the unique least squares solution is

$$\hat{x} = (A^T A)^{-1} A^T y. \quad (1.208b)$$

When  $A$  is square, the invertibility of  $A^T A$  implies  $y \in \mathcal{R}(A)$  and the least square solution simplifies to the exact solution  $\hat{x} = A^{-1}y$ .

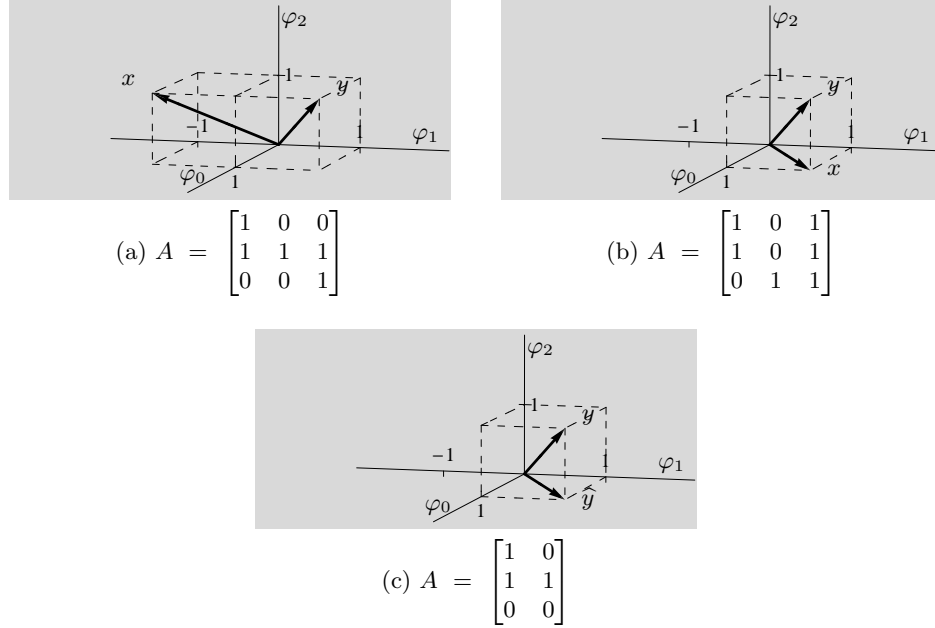
When  $A^T A$  is not invertible ( $\text{rank}(A) < N$ ), the minimization of  $\|y - A\hat{x}\|_2$  does not have a unique solution, so we additionally minimize  $\|\hat{x}\|_2$ . When  $AA^T$  is invertible ( $\text{rank}(A) = M$ ), this solution is

$$\hat{x} = A^T (AA^T)^{-1} y. \quad (1.208c)$$

The solutions (1.208b) and (1.208c) show the two forms of the *pseudoinverse* of  $A$  for  $\text{rank}(A) = \min(M, N)$ . Multiplication by the pseudoinverse solves the least squares problem for the case of  $\text{rank}(A) < \min(M, N)$  as well; the pseudoinverse is conveniently expressed using the singular value decomposition of  $A$  below. Figure 1.37 illustrates the discussion.

**Eigenvalues, Eigenvectors, and Spectral Decomposition** A number  $\lambda$  and nonzero vector  $v$  are called an *eigenvalue* and *eigenvector* of a square matrix  $A$  (also, an *eigenpair*) when

$$Av = \lambda v, \quad (1.209)$$



**Figure 1.37:** Illustration of solutions to  $Ax = y$  in  $\mathbb{R}^3$ , with  $y = [1 \ 1 \ 1]^T$ . (a) Unique solution:  $y \in \mathcal{R}(A)$  and the columns of  $A$  are linearly independent. The unique solution is  $x = [1 \ -1 \ 1]^T$ . (b) Infinitely many solutions:  $y \in \mathcal{R}(A)$  and the columns of  $A$  are not linearly independent. One of the possible solutions is  $x = [1 \ 1 \ 0]^T$ . (c) No solution:  $y \notin \mathcal{R}(A)$  and the columns of  $A$  are linearly independent. The unique approximate solution of minimum 2-norm minimizing the error is  $\hat{x} = [1 \ 0]^T$ ;  $\hat{y} = [1 \ 1 \ 0]^T$ .

as seen for general linear operators in (1.53). The eigenvalues are the roots of the *characteristic polynomial*  $\det(xI - A)$ . When all eigenvalues of  $A$  are real,  $\lambda_{\max}(A)$  denotes the largest eigenvalue and  $\lambda_{\min}(A)$  the smallest eigenvalue. Especially when the eigenvalues are nonnegative, it is conventional to list them in nonincreasing order,  $\lambda_0(A) \geq \lambda_1(A) \geq \dots \geq \lambda_{N-1}(A)$ .

If an  $N \times N$  matrix  $A$  has  $N$  linearly independent eigenvectors, then it can be written as

$$A = V\Lambda V^{-1}, \quad (1.210a)$$

where  $\Lambda$  is a diagonal matrix containing the eigenvalues of  $A$  along the diagonal and  $V$  contains the eigenvectors of  $A$  as its columns. This is called the *spectral theorem*. Since the eigenvectors form a basis in this case, a vector  $x$  can be written as a linear combination of eigenvectors  $x = \sum_{k=0}^{N-1} \alpha_k v_k$ , and

$$Ax = A \left( \sum_{k=0}^{N-1} \alpha_k v_k \right) \stackrel{(a)}{=} \sum_{k=0}^{N-1} \alpha_k (Av_k) \stackrel{(b)}{=} \sum_{k=0}^{N-1} (\alpha_k \lambda_k) v_k, \quad (1.210b)$$

where (a) follows from the linearity of  $A$ ; and (b) from (1.209). Expressions (1.210a)

and (1.210b) are both *diagonalizations*. The first shows  $V^{-1}AV$  is a diagonal matrix; the second shows that expressing the input to operator  $A$  using the coordinates specified by the eigenvectors of  $A$  makes the action of  $A$  diagonal. Combining properties of the determinant that we have seen earlier with (1.210a),

$$\det(A) = \det(V\Lambda V^{-1}) = \det(VV^{-1})\det(\Lambda) = \prod_{k=0}^{N-1} \lambda_k. \quad (1.211)$$

The conclusion  $\det(A) = \prod_{k=0}^{N-1} \lambda_k$  holds even for matrices without full sets of eigenvectors, as long as eigenvalues are counted with multiplicities.

The *trace* is defined for square matrices as the sum of the diagonal entries. The trace of a product is invariant to cyclic permutation of the factors, for example  $\text{tr}(ABC) = \text{tr}(BCA) = \text{tr}(CAB)$ . It follows that the trace is invariant to similarity transformations:  $\text{tr}(BAB^{-1}) = \text{tr}(AB^{-1}B) = \text{tr}(A)$ . The trace is given by the sum of eigenvalues (counted with multiplicities),

$$\text{tr}(A) = \sum_{k=0}^{N-1} \lambda_k, \quad (1.212)$$

which is justified by (1.210a) for diagonalizable  $A$ .

**Singular Value Decomposition** *Singular value decomposition (SVD)* provides a diagonalization that applies to any rectangular or square matrix. An  $M \times N$  real or complex matrix  $A$  can be factored as follows:

$$A = U\Sigma V^*, \quad (1.213)$$

where  $U$  is an  $M \times M$  unitary matrix,  $V$  is an  $N \times N$  unitary matrix, and  $\Sigma$  is an  $M \times N$  matrix with nonnegative real values  $\{\sigma_k\}_{k=0}^{\min(M,N)-1}$  called *singular values* on the main diagonal and zeros elsewhere. The columns of  $U$  are called *left singular vectors* and the columns of  $V$  are called *right singular vectors*. As for eigenvalues,  $\sigma_{\max}(A)$  denotes the largest singular value and  $\sigma_{\min}(A)$  the smallest singular value. Also as for eigenvalues, it is conventional to list singular values in nonincreasing order,  $\sigma_{\max}(A) = \sigma_0(A) \geq \sigma_1(A) \geq \dots \geq \sigma_{N-1}(A) = \sigma_{\min}(A)$ . The number of nonzero singular values is the rank of  $A$ . The pseudoinverse of  $A$  is

$$A^\dagger = V\Sigma^\dagger U^* \quad (1.214)$$

where  $\Sigma^\dagger$  is the  $N \times M$  matrix with  $1/\sigma_k$  in the  $(k, k)$  position for each nonzero singular value and zeros elsewhere.

The following fact relates singular value and eigendecompositions (see also Exercise 1.67): Using the singular value decomposition (1.213),

$$\begin{aligned} AA^* &= (U\Sigma V^*)(V\Sigma^* U^*) = U\Sigma^2 U^*, \\ A^*A &= (V\Sigma^* U^*)(U\Sigma V^*) = V\Sigma^2 V^*, \end{aligned}$$

so the squares of the singular values of  $A$  are the nonzero eigenvalues of  $AA^*$  and  $A^*A$ , that is,

$$\sigma^2(A) = \lambda(AA^*) = \lambda(A^*A), \quad \text{for } \lambda \neq 0. \quad (1.215)$$

**Matrix Norms** Norms on matrices must satisfy the conditions in Definition 1.9. Many commonly-used norms on  $M \times N$  matrices are *operator norms* induced by norms on  $M$ - and  $N$ -dimensional vectors, as in Definition 1.18. Using the vector norms defined in (1.35a) and (1.35b),

$$\|A\|_{p,q} = \sup_{\|x\|_p=1} \|Ax\|_q, \quad (1.216)$$

and  $\|A\|_{p,p}$  is denoted  $\|A\|_p$ . A few of these simplify as follows:

$$\begin{aligned} \|A\|_1 &= \|A\|_{1,1} = \max_{0 \leq j \leq N-1} \sum_{i=0}^{M-1} |A_{i,j}|, \\ \|A\|_{1,2} &= \max_{0 \leq j \leq N-1} \left( \sum_{i=0}^{M-1} |A_{i,j}|^2 \right)^{1/2}, \\ \|A\|_{1,\infty} &= \max_{0 \leq i \leq M-1, 0 \leq j \leq N-1} |A_{i,j}|, \\ \|A\|_2 &= \|A\|_{2,2} = \sigma_{\max}(A) = \sqrt{\lambda_{\max}(A^*A)}, \\ \|A\|_{2,\infty} &= \max_{0 \leq i \leq M-1} \left( \sum_{j=0}^{N-1} |A_{i,j}|^2 \right)^{1/2}, \\ \|A\|_\infty &= \|A\|_{\infty,\infty} = \max_{0 \leq i \leq M-1} \sum_{j=0}^{N-1} |A_{i,j}|. \end{aligned}$$

The most common matrix norm that is not an operator norm is the *Frobenius norm*:

$$\|A\|_F = \sqrt{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} |A_{i,j}|^2} = \sqrt{\text{tr}(AA^*)}. \quad (1.217)$$

### 1.B.2 Special Matrices

**Unitary and Orthogonal Matrices** A square matrix  $U$  is called *unitary* when it satisfies

$$U^*U = UU^* = I. \quad (1.218)$$

Its inverse  $U^{-1}$  equals its Hermitian transpose  $U^*$ . A real unitary matrix satisfies

$$U^T U = U U^T = I, \quad (1.219)$$

and is called *orthogonal*.<sup>27</sup>

Unitary matrices preserve norms for all complex vectors,

$$\|Ux\| = \|x\|,$$

and more generally preserve inner products,

$$\langle Ux, Uy \rangle = \langle x, y \rangle.$$

Each eigenvalue of a unitary matrix has unit modulus, and all its eigenvectors are orthogonal. Each eigenvalue of an orthogonal matrix is  $\pm 1$  or part of a complex conjugate pair  $e^{\pm j\theta}$ . From (1.178a), its condition number is  $\kappa(U) = 1$ .

<sup>27</sup>It is sometimes a source of confusion that an orthogonal matrix has *orthonormal* (not merely orthogonal) columns (or rows).

**Rotations and Rotoinversions** From (1.219), the determinant of an orthogonal matrix satisfies  $(\det(U))^2 = 1$ . When  $\det(U) = 1$ , the orthogonal matrix is called a *rotation*; when  $\det(U) = -1$ , it is called an *improper rotation* or *rotoinversion*. In  $\mathbb{R}^2$ , a rotation is always of the form

$$\begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}, \quad (1.220a)$$

and a rotoinversion is always of the form

$$\begin{bmatrix} \cos \theta & \sin \theta \\ \sin \theta & -\cos \theta \end{bmatrix}. \quad (1.220b)$$

The rotoinversion can be interpreted as a composition of a rotation and a reflection of one coordinate. In  $\mathbb{R}^N$ , a rotation can always be written as a product of  $N(N-1)/2$  matrices that each performs a planar rotation in one pair of coordinates. For example, any rotation in  $\mathbb{R}^3$  can be written as

$$\begin{bmatrix} \cos \theta_{01} & -\sin \theta_{01} & 0 \\ \sin \theta_{01} & \cos \theta_{01} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta_{02} & 0 & -\sin \theta_{02} \\ 0 & 1 & 0 \\ \sin \theta_{02} & 0 & \cos \theta_{02} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_{12} & -\sin \theta_{12} \\ 0 & \sin \theta_{12} & \cos \theta_{12} \end{bmatrix}.$$

A general rotoinversion can be written similarly with one planar rotation replaced by a planar rotoinversion.

**Hermitian, Symmetric, and Normal Matrices** A *Hermitian* matrix is equal to its adjoint:

$$A = A^*. \quad (1.221a)$$

Such a matrix must be square and is also called *self-adjoint*. A real Hermitian matrix is equal to its transpose,

$$A = A^T, \quad (1.221b)$$

and is called *symmetric*. The 2-norm of a Hermitian matrix is

$$\|A\|_2 = |\lambda_{\max}|. \quad (1.222)$$

All the eigenvalues of a Hermitian matrix are real. All the eigenvectors corresponding to distinct eigenvalues are orthogonal. A Hermitian matrix can be diagonalized as

$$A = U\Lambda U^*, \quad (1.223a)$$

where  $U$  is a unitary matrix with eigenvectors of  $A$  as columns, and  $\Lambda$  the diagonal matrix of corresponding eigenvalues; this is the spectral theorem for Hermitian matrices. For the case of  $A$  real (symmetric),  $U$  is real (orthogonal); thus,

$$A = U\Lambda U^T. \quad (1.223b)$$

Equation (1.223a) further means that any Hermitian matrix can be factored as  $A = QQ^*$ , with  $Q = U\sqrt{\Lambda}$ . Its condition number is given in (1.178b).

A matrix  $A$  is called *normal* when it satisfies  $A^*A = AA^*$ ; in words, it commutes with its Hermitian transpose. Hermitian matrices are obviously normal. A matrix is normal if and only if it can be unitarily diagonalized as in (1.223a).

For a normal (Hermitian) matrix  $A$ , for each eigenvalue  $\lambda_k$  there is a singular value  $\sigma_k = |\lambda_k|$ , where the singular values are listed in nonincreasing order, but the eigenvalues may not be.

**Positive Definite Matrices** A Hermitian matrix  $A$  is called *positive semidefinite* when, for all nonzero vectors  $x$ , the following is satisfied:

$$x^*Ax \geq 0. \quad (1.224)$$

This is also written as  $A \geq 0$ . If furthermore (1.224) holds with strict inequality,  $A$  is called *positive definite*, which is written as  $A > 0$ . When a Hermitian matrix  $A$  has smallest and largest eigenvalues of  $\lambda_{\min}$  and  $\lambda_{\max}$ , the matrices  $\lambda_{\max}I - A$  and  $A - \lambda_{\min}I$  are positive semidefinite:

$$\lambda_{\min}I \leq A \leq \lambda_{\max}I. \quad (1.225)$$

All eigenvalues of a positive definite matrix are positive. For any positive definite matrix  $A$ , there exists a nonsingular matrix  $W$  such that  $A = W^*W$ , where  $W$  is a matrix generalization of the square root of  $A$ . One possible way to choose such a square root is to diagonalize  $A$  as

$$A = Q\Lambda Q^*, \quad (1.226)$$

and, since all the eigenvalues are positive, choose  $W^* = Q\sqrt{\Lambda}$ , where the square root is applied to each diagonal element of  $\Lambda$ .

**Circulant Matrices** A (right) *circulant* matrix is a matrix where each row is obtained by a (right) circular shift of the previous row:

$$C = \begin{bmatrix} c_0 & c_{N-1} & \dots & c_1 \\ c_1 & c_0 & \dots & c_2 \\ \vdots & \vdots & \ddots & \vdots \\ c_{N-1} & c_{N-2} & \dots & c_0 \end{bmatrix}. \quad (1.227)$$

A circulant matrix is diagonalized by the DFT matrix (2.161), as we will see in (2.177). Since the DFT matrix is unitary, all the eigenvectors of a circulant matrix are orthonormal.

**Toeplitz Matrices** A *Toeplitz*  $T$  matrix is a matrix whose entry  $T_{ki}$  depends only on the value of  $k - i$ . A Toeplitz matrix is thus constant along diagonals:

$$T = \begin{bmatrix} t_0 & t_1 & t_2 & \cdots & t_{N-1} \\ t_{-1} & t_0 & t_1 & \cdots & t_{N-2} \\ t_{-2} & t_{-1} & t_0 & \cdots & t_{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{-N+1} & t_{-N+2} & t_{-N+3} & \cdots & t_0 \end{bmatrix}. \quad (1.228)$$

A matrix in which blocks follow the form above is called *block Toeplitz*.

**Band Matrices** A *band* or *banded* matrix is a square matrix with nonzero entries only in a “band” around the main diagonal. The band need not be symmetric; there may be  $N_r$  occupied diagonals on the right side and  $N_\ell$  on the left side. For example, a  $5 \times 5$  matrix with  $N_r = 2$  and  $N_\ell = 1$  is of the form:

$$B = \begin{bmatrix} b_{00} & b_{01} & b_{02} & 0 & 0 \\ b_{10} & b_{11} & b_{12} & b_{13} & 0 \\ 0 & b_{21} & b_{22} & b_{23} & b_{24} \\ 0 & 0 & b_{32} & b_{33} & b_{34} \\ 0 & 0 & 0 & b_{43} & b_{44} \end{bmatrix}. \quad (1.229)$$

Many sets of special matrices are subsets of the band matrices. For example, diagonal matrices have  $N_r = N_\ell = 0$ , tridiagonal have  $N_r = N_\ell = 1$ , upper-triangular have  $N_\ell = 0$ , and lower-triangular have  $N_r = 0$ .

Square matrices have a well-defined “main antidiagonal” running from lower-left corner to upper-right corner. An *antidiagonal matrix* has nonzero entries only in the main antidiagonal. A useful matrix is the *unit antidiagonal matrix*, which has 1s in the main antidiagonal.

**Vandermonde Matrices** A *Vandermonde matrix* is a matrix of the form:

$$V = \begin{bmatrix} 1 & \alpha_0 & \alpha_0^2 & \cdots & \alpha_0^{N-1} \\ 1 & \alpha_1 & \alpha_1^2 & \cdots & \alpha_1^{N-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha_{M-1} & \alpha_{M-1}^2 & \cdots & \alpha_{M-1}^{N-1} \end{bmatrix}. \quad (1.230)$$

When  $M = N$ , the determinant of the matrix is

$$\det(V) = \prod_{0 \leq i < j \leq N-1} (\alpha_i - \alpha_j). \quad (1.231)$$

Many useful concepts in sequence processing use Vandermonde matrices, such as the DFT matrix introduced in (2.161a).

## 1.C Elements of Probability

This appendix reviews basic concepts in the theory of probability, with an emphasis on continuous random variables. See [11] for a thorough but elementary introduction or [65] for an introduction with more mathematical sophistication.

### 1.C.1 Basic Definitions

**Probabilistic Models** A *probability law*  $P(\cdot)$  assigns probabilities to *events*, which are subsets of the *outcomes* of an experiment. The set of all outcomes is called the *sample space* and denoted  $\Omega$ . A probability law satisfies the following axioms:

- (i) *Nonnegativity.*  $P(A) \geq 0$  for every event  $A$ .
- (ii) *Additivity.* If  $A$  and  $B$  are disjoint events, then  $P(A \cup B) = P(A) + P(B)$ ; this additivity extends to countable unions of disjoint events.
- (iii) *Normalization.*  $P(\Omega) = 1$ .

The *conditional probability* of event  $A$ , given event  $B$  with  $P(B) > 0$ , is defined as

$$P(A|B) = \frac{P(A \cap B)}{P(B)}.$$

Conditioning on  $B$  is a restriction of the sample space to  $B$ , with rescaling of probabilities such that  $P(\cdot|B)$  satisfies the normalization axiom and thus is a probability law. If events  $A$  and  $B$  both have positive probability, then writing  $P(A \cap B) = P(A|B)P(B)$  and  $P(A \cap B) = P(B|A)P(A)$  yields *Bayes' Rule*:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}.$$

Events  $A$  and  $B$  are called *independent* when  $P(A \cap B) = P(A)P(B)$ .

**Continuous Random Variables** A real, continuous random variable  $x$  has a *probability density function (PDF)*  $f_x$  defined on the real line such that

$$P(x \in A) = \int_A f_x(t) dt \quad (1.232a)$$

is the probability that  $x$  falls in the set  $A \subset \mathbb{R}$ .<sup>28</sup> The *cumulative distribution function (CDF)* of  $x$  is

$$F_x(t) = P(x \leq t) = \int_{-\infty}^t f_x(s) ds. \quad (1.232b)$$

<sup>28</sup>Formally,  $x: \Omega \rightarrow \mathbb{R}$ , and  $\{\omega \in \Omega \mid x(\omega) \in A\}$  must be an event. There are technical subtleties in the functions  $f_x$  and sets  $A$  that should be allowed. It is adequate to assume that  $f_x$  has a countable number of discontinuities and that  $A$  is a countable union of intervals [65]. We refer the reader to the footnote on page 23 for our philosophy on this type of mathematical technicality.



Since probabilities are nonnegative, we must have  $f_x(t) \geq 0$  for all  $t \in \mathbb{R}$ . Since  $x$  takes some real value,

$$\int_{-\infty}^{\infty} f_x(t) dt = 1.$$

Elementary properties of the CDF include

$$\lim_{t \rightarrow -\infty} F_x(t) = 0 \quad \text{and} \quad \lim_{t \rightarrow \infty} F_x(t) = 1,$$

and that

$$\frac{d}{dt} F_x(t) = f_x(t) \quad \text{where the derivative exists.}$$

By calling  $f_x$  a function, we are excluding Dirac delta components from  $f_x$ ; the CDF  $F_x$  is then continuous because it is the integral of the PDF. Allowing Dirac components in  $f_x$  would introduce jumps in the CDF. This is necessary for describing discrete or mixed random variables.

**Expectation, Moments, and Variance** The *expectation* of a function  $g(x)$  is defined as

$$E[g(x)] = \int_{-\infty}^{\infty} g(t) f_x(t) dt. \quad (1.233a)$$

In particular,  $E[x^k]$  is called the *kth moment*. The zeroth moment must be 1, and other moments do not always exist. The first moment is called the *mean*, and the *variance* is obtained from the first and second moments:

$$\text{var}(x) = E[(x - E[x])^2] = E[x^2] - (E[x])^2. \quad (1.233b)$$

Variance is nonnegative. The expectation is linear in that

$$E[\alpha_0 g_0(x) + \alpha_1 g_1(x)] = \alpha_0 E[g_0(x)] + \alpha_1 E[g_1(x)] \quad (1.233c)$$

for any constants  $\alpha_0$  and  $\alpha_1$  and any functions  $g_0$  and  $g_1$ . From this it follows that

$$\text{var}(\alpha_0 x + \alpha_1) = \alpha_0^2 \text{var}(x) \quad (1.233d)$$

for any constants  $\alpha_0$  and  $\alpha_1$ .

Random variables  $x$  and  $y$  are said to have the same distribution when  $E[g(x)] = E[g(y)]$  for any function  $g$ . This requires their CDFs to be equal, though their PDFs may differ at a countable number of points.<sup>29</sup>

<sup>29</sup>This is analogous to equality in  $\mathcal{L}^p$  for  $1 \leq p < \infty$ : equality of CDFs  $F_X$  and  $F_Y$  implies  $\|f_X - f_Y\|_{\mathcal{L}^p} = 0$  for any  $p \in [1, \infty)$ .

**Jointly-Distributed Random Variables** Real, continuous random variables  $x$  and  $y$  have a *joint PDF*  $f_{x,y}$  defined such that

$$P((x, y) \in A) = \iint_A f_{x,y}(s, t) dt ds \quad (1.234a)$$

is the probability that  $(x, y)$  falls in  $A \subset \mathbb{R}^2$  and a *joint CDF*

$$F_{x,y}(s, t) = P(x \leq s, y \leq t) = \int_{-\infty}^s \int_{-\infty}^t f_{x,y}(u, v) dv du. \quad (1.234b)$$

The *marginal PDF* of  $x$  is

$$f_x(s) = \int_{-\infty}^{\infty} f_{x,y}(s, t) dt, \quad (1.234c)$$

and the *marginal CDF* of  $x$  is

$$F_x(s) = \lim_{t \rightarrow \infty} F_{x,y}(s, t). \quad (1.234d)$$

The *conditional PDF* of  $x$  given  $y$  is defined as

$$f_{x|y}(s | t) = \frac{f_{x,y}(s, t)}{f_y(t)} \quad \text{for } t \text{ such that } f_y(t) \neq 0. \quad (1.234e)$$

The *conditional expectation* is defined with the conditional PDF:

$$E[g(x) | y = t] = \int_{-\infty}^{\infty} g(s) f_{x|y}(s | t) ds. \quad (1.234f)$$

When  $f_{x,y}$  is separable as

$$f_{x,y}(s, t) = f_x(s)f_y(t) \quad (1.234g)$$

for PDFs  $f_x$  and  $f_y$ , the random variables  $x$  and  $y$  are called *independent*. An immediate ramification of independence is that  $f_{x|y}(s | t) = f_x(s)$  for every  $t$  such that  $f_y(t) \neq 0$ . These definitions extend to any number of random variables, with some subtleties for infinite collections.

A complex random variable has real and imaginary parts that are jointly distributed real random variables. A random vector has components that are jointly distributed scalar random variables. The mean of an  $N$ -dimensional random vector  $x$  is a vector  $\mu = E[x] \in \mathbb{R}^N$ . The *covariance matrix* is defined as  $E[(x - \mu)(x - \mu)^T]$ .

## 1.C.2 Standard Distributions

**Uniform Random Variables** For any real numbers  $a$  and  $b$  with  $a < b$ , a random variable with PDF

$$f_x(t) = \begin{cases} 1/(b-a), & \text{for } t \in [a, b]; \\ 0, & \text{otherwise} \end{cases} \quad (1.235a)$$

is called *uniform on*  $[a, b]$ . This is denoted  $x \sim \mathcal{U}(a, b)$ . Simple computations yield

$$F_x(t) = \begin{cases} 0, & \text{for } t < a; \\ (t-a)/(b-a), & \text{for } t \in [a, b]; \\ 1, & \text{for } t > b, \end{cases} \quad (1.235b)$$

$E[x] = (a+b)/2$ , and  $\text{var}(x) = (b-a)^2/12$ .

**Gaussian Random Variables and Vectors** For any real  $\mu$  and positive  $\sigma$ , a random variable with PDF

$$f_x(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(t-\mu)^2/\sigma^2} \quad (1.236)$$

is called *Gaussian* or *normal* with mean  $\mu$  and variance  $\sigma^2$ . This is denoted  $x \sim \mathcal{N}(\mu, \sigma^2)$ . When  $\mu = 0$  and  $\sigma = 1$ , the random variable is called *standard*. There is no elementary expression for the CDF of a Gaussian random variable.

For any  $\mu \in \mathbb{R}^N$  and symmetric, positive definite  $\Sigma \in \mathbb{R}^{N \times N}$ , a random vector  $x = [x_0, x_1, \dots, x_{N-1}]^T$  with joint PDF

$$f_x(t) = \frac{1}{(2\pi)^{N/2}(\det(\Sigma))^{1/2}} e^{-\frac{1}{2}(t-\mu)^T \Sigma^{-1}(t-\mu)} \quad (1.237)$$

is called (*jointly*) *Gaussian* or *multivariate normal* with mean  $\mu$  and covariance  $\Sigma$ . This is denoted  $x \sim \mathcal{N}(\mu, \Sigma)$ .

Gaussianity is invariant to affine transformations: when  $x$  is jointly Gaussian,  $Ax + b$  is also jointly Gaussian for any constant matrix  $A \in \mathbb{R}^{M \times N}$  of rank  $M$  and constant vector  $b \in \mathbb{R}^M$ ,<sup>30</sup> the new mean is  $A\mu + b$  and the new covariance matrix is  $A\Sigma A^T$ .

The marginal distributions and conditional distributions are jointly Gaussian also. Partition  $x$ ,  $\mu$ , and  $\Sigma$  (in a dimensionally-compatible manner) as

$$x = \begin{bmatrix} y \\ z \end{bmatrix}, \quad \mu = \begin{bmatrix} \mu_y \\ \mu_z \end{bmatrix}, \quad \text{and} \quad \Sigma = \begin{bmatrix} \Sigma_y & \Sigma_{y,z} \\ \Sigma_{z,y} & \Sigma_z \end{bmatrix}.$$

The symmetry of  $\Sigma$  implies  $\Sigma_y = \Sigma_y^T$ ,  $\Sigma_z = \Sigma_z^T$ , and  $\Sigma_{y,z} = \Sigma_{z,y}^T$ . The marginal distribution of  $y$  is jointly Gaussian with mean  $\mu_y$  and covariance  $\Sigma_y$ , and the conditional distribution of  $y$  given  $z = t$  is jointly Gaussian with mean  $\mu_y + \Sigma_{y,z}\Sigma_z^{-1}(t - \mu_z)$  and covariance  $\Sigma_y - \Sigma_{y,z}\Sigma_z^{-1}\Sigma_{z,y}$ . These and other properties of jointly Gaussian vectors are developed in Exercise 1.70.

Gaussian random variables are very common in modeling physical phenomena because they arise from the accumulation of a large number of small, independent, random effects. This is made precise by the *central limit theorem*, a simple version of which is as follows:

<sup>30</sup>Some authors require the covariance matrix of a jointly Gaussian vector to be positive semidefinite rather than positive definite. In this case, the rank condition on  $A$  can be removed, and the PDF does not necessarily exist.

Let  $x_1, x_2, \dots$  be independent, identically-distributed (i.i.d.) random variables with mean  $\mu$  and variance  $\sigma^2$ . For each  $n \in \mathbb{Z}^+$ , define a shifted and scaled version of the sample mean of  $\{x_k\}_{k=1}^n$ :

$$z_n = \frac{\sqrt{n}}{\sigma} \left( \left( \frac{1}{n} \sum_{k=1}^n x_k \right) - \mu \right). \quad (1.238a)$$

These random variables converge in distribution to a standard normal random variable  $z$ :

$$\lim_{n \rightarrow \infty} F_{z_n}(t) = F_z(t) \quad \text{for all } t \in \mathbb{R}. \quad (1.238b)$$

Similar results hold under conditions that allow weak dependence of the variables.

### 1.C.3 Estimation

Estimation is the process of forming estimates of parameters of interest from observations that are probabilistically related to the parameters. Bayesian and non-Bayesian (classical) techniques are distinguished by whether the parameters are considered to be random variables; observations are random in either case. For simplicity, generalities below are stated for a continuous scalar parameter and continuous scalar observations. Some examples demonstrate extensions to vectors.

**Bayesian Estimation** In *Bayesian estimation*, the parameter of interest is assumed to be a random variable  $x$ . Its distribution is called the *prior distribution* to emphasize that it describes  $x$  without use of observations. The conditional distribution of the observation  $y$ , given the parameter  $x$ , follows a distribution  $f_{y|x}$  called the *likelihood*. After observing  $y = t$ , Bayes' Rule specifies the *posterior distribution* of  $x$  to be

$$f_{x|y}(s|t) = \frac{f_x(s)f_{y|x}(t|s)}{f_y(t)} = \frac{f_x(s)f_{y|x}(t|s)}{\int_{-\infty}^{\infty} f_x(s)f_{y|x}(t|s) ds}.$$

Bayesian estimators are derived by using the posterior distribution to optimize a criterion of interest to find the best function  $\hat{x} = g(y)$ .

A common performance criterion is the *mean-squared error* (MSE)  $E[(x - \hat{x})^2]$ . In the trivial case of having no observation available,  $\hat{x}$  is simply a constant  $c$ . The MSE is minimized by  $c = E[x]$ . This is verified through the following expansion:

$$E[(x - c)^2] \stackrel{(a)}{=} \text{var}(x - c) + (E[x - c])^2 \stackrel{(b)}{=} \text{var}(x) + (E[x] - c)^2,$$

where (a) uses (1.233b); and (b) uses (1.233c) and (1.233d). When  $y = t$  has been observed,  $x$  is conditionally distributed as  $f_{x|\{y=t\}}$  and the MSE is minimized by

$$\hat{x}_{\text{MMSE}}(t) = E[x|y = t].$$

EXAMPLE 1.64 (BAYESIAN MMSE ESTIMATION: GAUSSIAN CASE) Let  $\mathbf{x}$  and  $\mathbf{y}$  be jointly Gaussian vectors, meaning that their concatenation  $[\mathbf{x}^T \ \mathbf{y}^T]^T$  has PDF of the form (1.237). Assume  $E[\mathbf{x}] = \mu_{\mathbf{x}}$  and  $E[\mathbf{y}] = \mu_{\mathbf{y}}$ , and write the covariance as

$$E \left[ \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \begin{bmatrix} \mathbf{x}^T & \mathbf{y}^T \end{bmatrix} \right] = \begin{bmatrix} \Sigma_{\mathbf{x}} & \Sigma_{\mathbf{x},\mathbf{y}} \\ \Sigma_{\mathbf{x},\mathbf{y}}^T & \Sigma_{\mathbf{y}} \end{bmatrix}$$

where  $\Sigma_{\mathbf{x}}$  is the covariance of  $\mathbf{x}$ ,  $\Sigma_{\mathbf{y}}$  is the covariance of  $\mathbf{y}$ , and  $\Sigma_{\mathbf{x},\mathbf{y}} = E[\mathbf{x}\mathbf{y}^T]$  is the cross covariance between  $\mathbf{x}$  and  $\mathbf{y}$ .

The conditional PDF of  $\mathbf{x}$  given  $\mathbf{y} = t$  is jointly Gaussian with mean  $\mu_{\mathbf{x}} + \Sigma_{\mathbf{x}\mathbf{y}}\Sigma_{\mathbf{y}}^{-1}(t - \mu_{\mathbf{y}})$  and covariance  $\Sigma_{\mathbf{x}} - \Sigma_{\mathbf{x},\mathbf{y}}\Sigma_{\mathbf{y}}^{-1}\Sigma_{\mathbf{y},\mathbf{x}}^T$  (see Appendix 1.C.2). Since the conditional mean minimizes MSE, we have

$$\hat{\mathbf{x}}_{\text{MMSE}}(t) = \mu_{\mathbf{x}} + \Sigma_{\mathbf{x},\mathbf{y}}\Sigma_{\mathbf{y}}^{-1}(t - \mu_{\mathbf{y}}). \quad (1.239)$$

Its resulting MSE is

$$E[\|\mathbf{x} - \hat{\mathbf{x}}_{\text{MMSE}}\|^2] = \text{tr}(\Sigma_{\mathbf{x}} - \Sigma_{\mathbf{x},\mathbf{y}}\Sigma_{\mathbf{y}}^{-1}\Sigma_{\mathbf{y},\mathbf{x}}^T).$$

Note that the same optimal estimator arises for general (non-Gaussian) distributions when the estimator is restricted to be linear; see Exercise 1.34.

As a special case, suppose  $\mathbf{y} = \mathbf{x} + \mathbf{z}$  where  $\mathbf{z} \sim \mathcal{N}(0, \sigma_z^2 I)$  and  $\mathbf{x}$  and  $\mathbf{z}$  are independent. We say that  $\mathbf{y}$  is an observation of  $\mathbf{x}$  with additive white Gaussian noise (AWGN). Then  $\mu_{\mathbf{y}} = \mu_{\mathbf{x}}$ ,  $\Sigma_{\mathbf{x},\mathbf{y}} = E[\mathbf{x}(\mathbf{x} + \mathbf{z})^T] = \Sigma_{\mathbf{x}}$ , and  $\Sigma_{\mathbf{y}} = E[(\mathbf{x} + \mathbf{z})(\mathbf{x} + \mathbf{z})^T] = \Sigma_{\mathbf{x}} + \sigma_z^2 I$ . The optimal estimator from observation  $\mathbf{y} = t$  and its performance simplify to

$$\hat{\mathbf{x}}_{\text{MMSE}}(t) = \mu_{\mathbf{x}} + \Sigma_{\mathbf{x}}(\Sigma_{\mathbf{x}} + \sigma_z^2 I)^{-1}(t - \mu_{\mathbf{x}}),$$

$$E[\|\mathbf{x} - \hat{\mathbf{x}}_{\text{MMSE}}\|^2] = \text{tr}(\Sigma_{\mathbf{x}} - \Sigma_{\mathbf{x}}(\Sigma_{\mathbf{x}} + \sigma_z^2 I)^{-1}\Sigma_{\mathbf{x}}).$$

Specializing further, suppose  $\mathbf{x}$  is scalar with mean zero and variance  $\sigma_x^2$ . Then

$$\hat{x}_{\text{MMSE}}(t) = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_z^2} t$$

and

$$E[\|x - \hat{x}_{\text{MMSE}}\|^2] = \sigma_x^2 \left( 1 - \frac{\sigma_x^2}{\sigma_x^2 + \sigma_z^2} \right).$$

**Classical Estimation** In *classical estimation*, the parameter of interest is treated as an unknown non-random quantity. The observation  $\mathbf{y}$  is random, with a *likelihood* (distribution)  $f_{\mathbf{y};\mathbf{x}}$  that depends on the parameter  $\mathbf{x}$ .<sup>31</sup> An estimator  $\hat{\mathbf{x}}(\mathbf{y})$  produces an estimate of  $\mathbf{x}$  from the observation. Since it is a function of the random variable  $\mathbf{y}$ , an estimate is also a random variable with a distribution that depends on  $\mathbf{x}$ . The dependence on  $\mathbf{x}$  is emphasized with a subscript in the following.

<sup>31</sup>Some authors write this as  $f_{\mathbf{y}|\mathbf{x}}$ , with the potential to confuse random and non-random quantities.

The *error* of an estimator is  $\hat{x}(y) - x$ , and the *bias* is the expected error:

$$b_x(\hat{x}(y)) = E_x[\hat{x}(y) - x] = E_x[\hat{x}(y)] - x.$$

An *unbiased* estimator has  $b_x(\hat{x}(y)) = 0$  for all  $x$ . The *mean-squared error* of an estimator is

$$E_x[|\hat{x}(y) - x|^2].$$

It depends on  $x$ , and it can be expanded as the sum of the variance and the square of the bias:

$$E_x[|\hat{x}(y) - x|^2] = \text{var}_x(\hat{x}(y)) + (b_x(\hat{x}(y)))^2.$$

Sometimes attention is limited to unbiased estimators, in which case the MSE is minimized by minimizing the variance of the estimator. This results in the *minimum-variance unbiased estimator* (MVUE).

EXAMPLE 1.65 (CLASSICAL MMSE ESTIMATION: GAUSSIAN CASE) Let  $x \in \mathbb{R}^N$  be a parameter of interest, and let  $y = Ax + z$  where  $A \in \mathbb{R}^{M \times N}$  is a known matrix and  $z \sim \mathcal{N}(0, \Sigma)$ . Since  $z$  has zero mean, the estimator  $\hat{x}(y) = By$  is unbiased whenever  $BA = I$ . Assuming  $BA = I$ , the MSE of the estimator is

$$E_x[\|B(Ax + z) - x\|^2] = E_x[\|Bz\|^2] = \text{tr}(B\Sigma B^T).$$

Since this MSE does not depend on  $x$ , it can be minimized through the choice of  $B$  to yield a valid estimator. The MSE is minimized by  $B = (A^T \Sigma^{-1} A)^{-1} A^T \Sigma^{-1}$ . The resulting MSE is  $\text{tr}((A^T \Sigma^{-1} A)^{-1})$ .

Note that  $A^T \Sigma^{-1} A$  must be invertible for the estimator above to exist. If  $\text{rank}(A) < N$ , it is hopeless to form an estimate of  $x$  without prior information; the component of  $x$  in the null space of  $A$  is unobserved.

As a special case, suppose  $\Sigma = \sigma_z^2 I$ . Then the optimal estimator simplifies to  $B = (A^T A)^{-1} A^T$ , the pseudoinverse of  $A$ .

## Chapter at a Glance

In this chapter, our goal was to find representations given by linear operators such that:

$$x = \Phi \tilde{\Phi}^* x.$$

After finding  $\Phi$  and  $\tilde{\Phi}$  such that  $\Phi \tilde{\Phi}^* = I$ , we call

$$\alpha = \tilde{\Phi}^* x, \quad x = \Phi \alpha = \Phi \tilde{\Phi}^* x,$$

a *decomposition* and a *reconstruction*, respectively. Also,  $\Phi \alpha$  is often called a *representation* of a signal. The elements of  $\alpha$  are called *expansion or transform coefficients* and include Fourier, wavelet, Gabor coefficients, as well as many others. We decompose signals to look into their properties in the *transform domain*. After analysis or manipulations such as compression, transmission, etc., we reconstruct the signal from its expansion coefficients. We studied cases distinguished by the properties of  $\Phi$ ; in finite dimensions,

- (i) If  $\Phi$  is square and nonsingular, then  $\Phi$  is a basis and  $\tilde{\Phi}$  is its dual basis.
- (ii) If  $\Phi$  is unitary, that is,  $\Phi \Phi^* = I$ , then  $\Phi$  is an orthonormal basis and  $\tilde{\Phi} = \Phi$ .
- (iii) If  $\Phi$  is rectangular and full rank, then  $\Phi$  is a frame and  $\tilde{\Phi}$  is its dual frame.
- (iv) If  $\Phi$  is rectangular and  $\Phi \Phi^* = I$ , then  $\Phi$  is a tight frame and  $\tilde{\Phi} = \Phi$ .

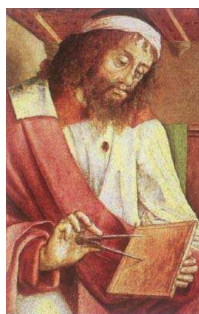
Which of these options we will choose depends entirely on our application and the criteria for designing such matrices (representations). In the book, these criteria are based on time-frequency considerations; we explore them in more detail in Chapter 6.

Signal Representations in Finite-Dimensional Spaces				
Property	Orth. basis	Biorth. basis	Tight frame	General frame
Expansion set	$\Phi = \{\varphi_k\}_{k=0}^{N-1}$ $\varphi_k \in \mathbb{C}^N$	$\Phi = \{\varphi_k\}_{k=0}^{N-1}$ $\tilde{\Phi} = \{\tilde{\varphi}_k\}_{k=0}^{N-1}$ $\varphi_k, \tilde{\varphi}_k \in \mathbb{C}^N$	$\Phi = \{\varphi_k\}_{k=0}^{M-1}$ $\varphi_k \in \mathbb{C}^N, M \geq N$	$\Phi = \{\varphi_k\}_{k=0}^{M-1}$ $\tilde{\Phi} = \{\tilde{\varphi}_k\}_{k=0}^{M-1}$ $\varphi_k, \tilde{\varphi}_k \in \mathbb{C}^N, M \geq N$
Structure	$\langle \varphi_i, \varphi_k \rangle = \delta_{i-k}$	$\langle \varphi_i, \tilde{\varphi}_k \rangle = \delta_{i-k}$	None	None
Expansion	$\sum_{k=0}^{N-1} \langle x, \varphi_k \rangle \varphi_k$	$\sum_{k=0}^{N-1} \langle x, \tilde{\varphi}_k \rangle \varphi_k$	$\sum_{k=0}^{M-1} \langle x, \varphi_k \rangle \varphi_k$	$\sum_{k=0}^{M-1} \langle x, \tilde{\varphi}_k \rangle \varphi_k$
Matrix view	$\Phi$ of size $N \times N$ $\Phi$ unitary $\Phi \Phi^* = \Phi^* \Phi = I$	$\Phi$ of size $N \times N$ $\Phi$ full rank $N$ $\Phi \tilde{\Phi}^* = I, \tilde{\Phi} = (\Phi^*)^{-1}$	$\Phi$ of size $N \times M$ rows of $\Phi$ orthogonal $\Phi \Phi^* = I$	$\Phi$ of size $N \times M$ $\Phi$ full rank $N$ $\Phi \tilde{\Phi}^* = I$
Norm pres.	Yes, $\ x\ ^2 = \sum_{k=0}^{N-1}  \langle x, \varphi_k \rangle ^2$	No	Yes, $\ x\ ^2 = \frac{1}{A} \sum_{k=0}^{M-1}  \langle x, \varphi_k \rangle ^2$	No
Successive approx.	Yes, $\hat{x}^{(k)} = \hat{x}^{(k-1)} + \langle x, \varphi_k \rangle \varphi_k$	No	No	No
Redundant	No	No	Yes	Yes

**Table 1.3:** Signal representations in finite-dimensional spaces.

## Historical Remarks

The choice of the title for this chapter requires at least some acknowledgment of the two mathematical giants figuring in it: Euclid and Hilbert.



Little is known about **Euclid (c. 300 BC)** apart from his writings. He was a Greek mathematician who lived and worked in Alexandria, Egypt. His book *Elements* [68] (in fact, 13 books), remains the most successful textbook in the history of mathematics. In it, he introduces and discusses many topics, most of which have taken hold in our consciousness as immutable truths, such as the principles of Euclidean geometry. He also has numerous results in number theory, including a simple proof that there are infinitely many prime numbers and a procedure for finding the greatest common divisor of two numbers. Moreover, it was Euclid who introduced the axiomatic method upon which all mathematical knowledge today is based.

**David Hilbert (1862–1943)** was a German mathematician, known for an axiomatization of geometry supplanting Euclid's five original axioms. He was one of the most universal mathematicians of all time, contributing towards knowledge in functional analysis, number theory and physics, among others. His efforts towards banishing theoretical uncertainties in mathematics ended in failure [174]: “Gödel demonstrated that any non-contradictory formal system, which was comprehensive enough to include at least arithmetic, cannot demonstrate its completeness by way of its own axioms.” At the turn of the 20th century, he produced a list of 23 unsolved problems, generally thought to be the most thoughtful and comprehensive such list ever. He worked closely with another famous mathematician, Minkowski, and had as students or assistants such illustrious names as Weyl, von Neumann, Courant and many others. He taught all his life, first at the University of Königsberg and then at the University of Göttingen, where he died in 1943. On his tombstone, one of his famous sayings is inscribed:



*Wir müssen wissen. Wir werden wissen.*<sup>32</sup>

## Further Reading

**Books and Textbooks** Below is a sample list of books/textbooks in which more information can be found about various topics we discussed in this chapter. Some of them are standard in the signal processing community and others we have used while writing this book.

Standard books on probability are by Bertsekas and Tsitsiklis [11] and Papoulis [110].

Many good reference texts exist on linear algebra, for example, Gantmacher [56] and Strang [139]. Good reviews are also provided by Kailath in [83] and Vaidyanathan [158]. Books by Kreyszig [93], Luenberger [98], Gohberg and Goldberg [59], and Young [176] provide details on abstract vector spaces. In particular, parts of our proof of the projection theorem, Theorem 1.26, follow [98] closely. Daubechies in [41] discusses Riesz bases, while more on frames can be found in [29, 90].

Parametrization of unitary matrices in various forms, such as using Givens rotations

<sup>32</sup>We must know. We will know.



or Householder building blocks are given in [158].

## Exercises with Solutions

### 1.1. Vector Space $\mathbb{C}^N$

Prove that:

- (i)  $\mathbb{C}^N$  is a vector space.
- (ii) The finite cross product of vector spaces,  $V = V_0 \times V_1 \times \cdots \times V_{N-1}$ , where  $V_0, V_1, \dots, V_{N-1}$  are each a vector space, is a vector space as well. The finite cross product is defined as the set of sequences  $x = [x_0 \ x_1 \ \cdots \ x_{N-1}]^T$  where  $x_0 \in V_0, \dots, x_{N-1} \in V_{N-1}$ .

*Solution:*

- (i) To prove that  $\mathbb{C}^N$  is a vector space, we need to check that the conditions stated in Definition 1.1 are satisfied. We prove the following for any  $x, y$  and  $z$  in  $\mathbb{C}^N$ . While these are rather trivial, we go through the details once.

#### 1. Commutativity:

$$\begin{aligned} x + y &= [x_0 \ x_1 \ \cdots \ x_{N-1}]^T + [y_0 \ y_1 \ \cdots \ y_{N-1}]^T \\ &= [x_0 + y_0 \ x_1 + y_1 \ \cdots \ x_{N-1} + y_{N-1}]^T \\ &= [y_0 + x_0 \ y_1 + x_1 \ \cdots \ y_{N-1} + x_{N-1}]^T \\ &= [y_0 \ y_1 \ \cdots \ y_{N-1}]^T + [x_0 \ x_1 \ \cdots \ x_{N-1}]^T = y + x. \end{aligned}$$

#### 2. Associativity:

$$\begin{aligned} (x + y) + z &= [x_0 + y_0 \ x_1 + y_1 \ \cdots \ x_{N-1} + y_{N-1}]^T + [z_0 \ z_1 \ \cdots \ z_{N-1}]^T \\ &= [(x_0 + y_0) + z_0 \ (x_1 + y_1) + z_1 \ \cdots \ (x_{N-1} + y_{N-1}) + z_{N-1}]^T \\ &= [x_0 + (y_0 + z_0) \ x_1 + (y_1 + z_1) \ \cdots \ x_{N-1} + (y_{N-1} + z_{N-1})]^T \\ &= [x_0 \ x_1 \ \cdots \ x_{N-1}]^T + [y_0 + z_0 \ y_1 + z_1 \ \cdots \ y_{N-1} + z_{N-1}]^T \\ &= x + (y + z), \end{aligned}$$

and

$$\begin{aligned} (\alpha\beta)x &= [(\alpha\beta)x_0 \ (\alpha\beta)x_1 \ \cdots \ (\alpha\beta)x_{N-1}]^T \\ &= [\alpha(\beta x_0) \ \alpha(\beta x_1) \ \cdots \ \alpha(\beta x_{N-1})]^T = \alpha(\beta x). \end{aligned}$$

#### 3. Distributivity: Follows similarly to the above two.

- 4. *Additive identity:* The element  $\mathbf{0} = [0 \ 0 \ \cdots \ 0]^T \in \mathbb{C}^N$  is unique, since all its components ( $0 \in \mathbb{C}$ ) are unique, and

$$\begin{aligned} x + \mathbf{0} &= [x_0 + 0 \ x_1 + 0 \ \cdots \ x_{N-1} + 0]^T \\ &= [0 + x_0 \ 0 + x_1 \ \cdots \ 0 + x_{N-1}]^T = \mathbf{0} + x \\ &= [x_0 \ x_1 \ \cdots \ x_{N-1}]^T = x. \end{aligned}$$

- 5. *Additive inverse:* The element  $(-x) = [-x_0 \ -x_1 \ \cdots \ -x_{N-1}]^T \in \mathbb{C}^N$  is unique, since  $(-x_i)$  for  $i = 0, 1, \dots, N-1$  are unique in  $\mathbb{C}$ , and

$$\begin{aligned} x + (-x) &= [x_0 + (-x_0) \ x_1 + (-x_1) \ \cdots \ x_{N-1} + (-x_{N-1})]^T \\ &= [(-x_0) + x_0 \ (-x_1) + x_1 \ \cdots \ (-x_{N-1}) + x_{N-1}]^T \\ &= (-x) + x = [0 \ 0 \ \cdots \ 0]^T = \mathbf{0}. \end{aligned}$$

6. *Multiplicative identity*: Follows similarly to additive identity.

- (ii) Thus  $\mathbb{C}^N$  is a vector space. Note that all the arguments above rely on the fact that  $\mathbb{C}$  itself is a vector space, and thus addition and scalar multiplication satisfy all the necessary properties. The finite cross product of vector spaces is a vector space. That is, let  $V_0, V_1, \dots, V_{N-1}$  be  $N$  vector spaces. We denote by  $V = V_0 \times V_1 \times \dots \times V_{N-1}$  the set of sequences  $x = [x_0 \ x_1 \ \dots \ x_{N-1}]^T$  where  $x_0 \in V_0, \dots, x_{N-1} \in V_{N-1}$ . Then  $V$  is a vector space with the following operations:

$$x + y = [x_0 \oplus_{V_0} y_0 \ \dots \ x_{N-1} \oplus_{V_{N-1}} y_{N-1}]^T$$

where “ $\oplus_{V_i}$ ” is the addition in  $V_i$  for  $i = 0, 1, \dots, N-1$ , and

$$\alpha x = [\alpha \odot_{V_0} x_0 \ \dots \ \alpha \odot_{V_{N-1}} x_{N-1}]^T$$

where “ $\odot_{V_i}$ ” is the scalar multiplication in  $V_i$  for  $i = 0, 1, \dots, N-1$ .

### 1.2. Vector Space $\ell^p(\mathbb{Z})$

Show that for any  $p \in [1, \infty)$ , the vectors in  $\mathbb{C}^{\mathbb{Z}}$  with finite  $\ell^p(\mathbb{Z})$  norm form a vector space. (*Hint*: Use Minkowski’s inequality (1.198a).)

*Solution*: Since  $\ell^p(\mathbb{Z})$  is a subset of  $\mathbb{C}^{\mathbb{Z}}$ , we are using the vector addition and scalar multiplication operations of  $\mathbb{C}^{\mathbb{Z}}$ . Thus, we need not check commutativity, associativity, distributivity, and multiplicative identity properties. The additive identity  $\mathbf{0} \in \mathbb{C}^{\mathbb{Z}}$  has  $\ell^p$  norm 0 and thus is in the subset under consideration. For any  $x \in \mathbb{C}^{\mathbb{Z}}$  with finite  $\ell^p$  norm,  $-x$  also has finite  $\ell^p$  norm, so the subset under consideration has the additive inverse property.

What remains is to show that, if  $x, y \in \ell^p(\mathbb{Z})$  and  $\alpha \in \mathbb{C}$ , then (i)  $\alpha x \in \ell^p(\mathbb{Z})$ ; and (ii)  $x + y \in \ell^p(\mathbb{Z})$ . To check (i), note that

$$\|\alpha x\|_p^p = \sum_{n=-\infty}^{\infty} |\alpha x_n|^p = |\alpha|^p \sum_{n=-\infty}^{\infty} |x_n|^p = |\alpha|^p \|x\|_p^p < \infty.$$

Property (ii) follows immediately from Minkowski’s inequality.

### 1.3. Incompleteness of $C([0, 1])$

Consider the sequence of functions  $\{x_k\}_{k \geq 2}$  on  $[0, 1]$  defined by

$$x_k(t) = \begin{cases} 0, & t \in [0, \frac{1}{2} - \frac{1}{k}); \\ k(t - \frac{1}{2}) + 1, & t \in [\frac{1}{2} - \frac{1}{k}, \frac{1}{2}); \\ 1, & t \in [\frac{1}{2}, 1]. \end{cases}$$

- Sketch the sequence of functions.
- Show that  $x_k$  is a Cauchy sequence under the  $\mathcal{L}^2$  norm.
- Show that  $x_k \rightarrow f$  under the  $\mathcal{L}^2$  norm for a discontinuous function  $f$ . Since  $f \notin C([0, 1])$ , this shows that  $C([0, 1])$  is not complete.

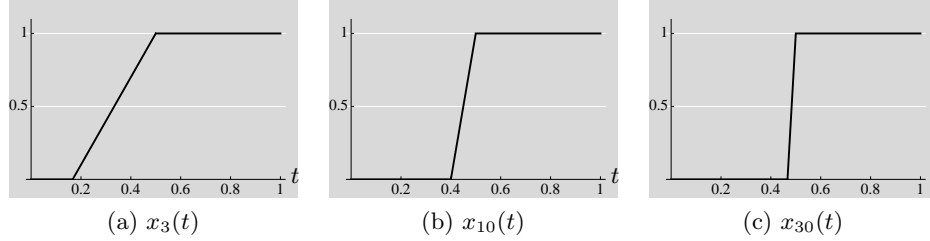
*Solution*:

- Figure E1.3-1 shows the plots for functions  $x_3, x_{10}$ , and  $x_{30}$ .
- $(x_n)_{n \geq 2}$  is a Cauchy sequence under the  $\mathcal{L}^2$  norm, if for any  $\epsilon > 0$  there exists an integer  $N_0 > 0$ , such that for any integers  $m, n \geq N_0$  the norm of the difference of  $x_n$  and  $x_m$  is bounded by  $\epsilon$ :  $\|x_m - x_n\|_2 \leq \epsilon$ .

Assume  $m \geq n$  and consider the function  $x_m - x_n$ :

$$x_m(t) - x_n(t) = \begin{cases} 0, & t \in [0, \frac{1}{2} - \frac{1}{n}) \cup [\frac{1}{2}, 1]; \\ -n(t - \frac{1}{2}) - 1, & t \in [\frac{1}{2} - \frac{1}{n}, \frac{1}{2} - \frac{1}{m}); \\ (m - n)(t - \frac{1}{2}), & t \in [\frac{1}{2} - \frac{1}{m}, \frac{1}{2}]. \end{cases}$$

Since our sequence starts at  $n = 2$ , for any  $\epsilon > 0$  we define  $N_0 = \max(2, \lceil \frac{4}{3\epsilon^2} \rceil)$ .

**Figure E1.3-1:** Example functions  $x_k(t)$ .

Then for any  $m \geq n \geq N_0$  the following holds:

$$\begin{aligned}
 \|x_m - x_n\|_2^2 &= \int_{\frac{1}{2} - \frac{1}{n}}^{\frac{1}{2}} (x_m - x_n)^2 dt \\
 &= \int_{\frac{1}{2} - \frac{1}{n}}^{\frac{1}{2} - \frac{1}{m}} (n(t - \frac{1}{2}) + 1)^2 dt + \int_{\frac{1}{2} - \frac{1}{m}}^{\frac{1}{2}} (m - n)^2 (t - \frac{1}{2})^2 dt \\
 &= \frac{(1 - \frac{n}{m})^3}{3n} + \frac{(m - n)^2}{3m^3} \\
 &= \frac{(m - n)^2}{3m^2 n} \leq \frac{(m - n)^2}{3m^2 n} = \frac{4}{3n} \leq \frac{4}{3N_0} \leq \epsilon^2.
 \end{aligned}$$

Hence,  $\|x_m - x_n\|_2 \leq \epsilon$ , and by definition,  $(x_n)$  is a Cauchy sequence under the  $\mathcal{L}^2$  norm.

(iii)

$$\lim_{n \rightarrow \infty} x_n(t) = f(t) = \begin{cases} 0, & t \in [0, \frac{1}{2}); \\ 1, & t \in [\frac{1}{2}, 1]. \end{cases}$$

Since  $f(t)$  has a discontinuity at  $t = \frac{1}{2}$ ,  $f \notin C([0, 1])$ .

#### 1.4. Orthogonal Projection to Span of an Orthonormal Set

Let  $\{\varphi_k\}_{k \in \mathcal{I}}$  be an orthonormal set. Prove that

$$Px = \sum_{k \in \mathcal{I}} \langle x, \varphi_k \rangle \varphi_k$$

is an orthogonal projection operator onto  $\overline{\text{span}}(\{\varphi_k\}_{k \in \mathcal{I}})$ .

*Solution:* The linearity of  $P$  follows from the distributivity and linearity in the first argument of the inner product. Verifying idempotency and self-adjointness establishes that  $P$  is an orthogonal projection operator, by application of Theorem 1.26.

(i) *Idempotency:*

$$\begin{aligned}
 P^2 x &= \sum_{m \in \mathcal{I}} \left\langle \sum_{k \in \mathcal{I}} \langle x, \varphi_k \rangle \varphi_k, \varphi_m \right\rangle \varphi_m \stackrel{(a)}{=} \sum_{m \in \mathcal{I}} \sum_{k \in \mathcal{I}} \langle \langle x, \varphi_k \rangle \varphi_k, \varphi_m \rangle \varphi_m \\
 &\stackrel{(b)}{=} \sum_{m \in \mathcal{I}} \sum_{k \in \mathcal{I}} \langle x, \varphi_k \rangle \langle \varphi_k, \varphi_m \rangle \varphi_m \stackrel{(c)}{=} \sum_{m \in \mathcal{I}} \sum_{k \in \mathcal{I}} \delta_{m-k} \langle x, \varphi_k \rangle \varphi_m \\
 &= \sum_{k \in \mathcal{I}} \langle x, \varphi_k \rangle \varphi_k = Px,
 \end{aligned}$$

where (a) follows from the distributivity of the inner product; (b) from the linearity in the first argument of the inner product; and (c) from the orthonormality of the set  $\{\varphi_k\}_{k \in \mathcal{I}}$ .

(ii) *Self-adjointness:*

$$\begin{aligned}
 \langle Px, y \rangle &= \left\langle \sum_{k \in \mathcal{I}} \langle x, \varphi_k \rangle \varphi_k, y \right\rangle \stackrel{(a)}{=} \sum_{k \in \mathcal{I}} \langle \langle x, \varphi_k \rangle \varphi_k, y \rangle \\
 &\stackrel{(b)}{=} \sum_{k \in \mathcal{I}} \langle x, \varphi_k \rangle \langle \varphi_k, y \rangle \stackrel{(c)}{=} \sum_{k \in \mathcal{I}} \langle \varphi_k, y \rangle \langle x, \varphi_k \rangle \\
 &\stackrel{(d)}{=} \sum_{k \in \mathcal{I}} \langle y, \varphi_k \rangle^* \langle x, \varphi_k \rangle \stackrel{(e)}{=} \sum_{k \in \mathcal{I}} \langle x, \langle y, \varphi_k \rangle \varphi_k \rangle \\
 &\stackrel{(f)}{=} \left\langle x, \sum_{k \in \mathcal{I}} \langle y, \varphi_k \rangle \varphi_k \right\rangle = \langle x, Py \rangle,
 \end{aligned}$$

where (a) follows from the distributivity of the inner product; (b) from the linearity in the first argument of the inner product; in (c) we just exchanged the order of the two inner products; (d) follows from the Hermitian symmetry of the inner product; (e) from the conjugate linearity in the second argument of the inner product; and (f) from the distributivity of the inner product.

#### 1.5. Legendre Polynomials

Consider the vectors  $1, t, t^2, t^3, \dots$  in the vector space  $\mathcal{L}^2([-1, 1])$ . Using Gram–Schmidt orthogonalization, find an orthonormal set with the same span.

*Solution:* Let  $x_k = t^{k-1}$  for  $k \in \mathbb{Z}^+$ . We initiate the Gram–Schmidt procedure with

$$\varphi_0 = \frac{x_0}{\|x_0\|} = \frac{1}{(\int_{-1}^1 1 dt)^{1/2}} = \frac{1}{\sqrt{2}}.$$

Continuing the Gram–Schmidt orthogonalization,

$$v_1 = \langle x_1, \varphi_0 \rangle \varphi_0 = \left( \int_{-1}^1 \frac{1}{\sqrt{2}} t dt \right) \frac{1}{\sqrt{2}} = 0,$$

$$\varphi_1 = \frac{x_1 - v_1}{\|x_1 - v_1\|} = \frac{t}{(\int_{-1}^1 t^2 dt)^{1/2}} = \sqrt{\frac{3}{2}} t,$$

$$v_2 = \langle x_2, \varphi_0 \rangle \varphi_0 + \langle x_2, \varphi_1 \rangle \varphi_1 = \left( \int_{-1}^1 \frac{1}{\sqrt{2}} t^2 dt \right) \frac{1}{\sqrt{2}} + \left( \int_{-1}^1 \sqrt{\frac{3}{2}} t \cdot t^2 dt \right) \sqrt{\frac{3}{2}} = \frac{1}{3},$$

$$\varphi_2 = \frac{x_2 - v_2}{\|x_2 - v_2\|} = \frac{t^2 - \frac{1}{3}}{(\int_{-1}^1 (t^2 - \frac{1}{3})^2 dt)^{1/2}} = \frac{3\sqrt{5}}{2\sqrt{2}} \left( t^2 - \frac{1}{3} \right).$$

This process can be continued to find an orthonormal set of arbitrary size. It can be proven by induction that

$$\varphi_{n+1} = \sqrt{\frac{2n+1}{2}} \cdot \frac{(-1)^n}{2^n n!} \cdot \frac{d^n}{dt^n} [(1-t^2)^n].$$

#### 1.6. Complexity of Matrix Multiplication and Strassen's Algorithm

In Example 1.56, we saw an algorithm allowing to multiply two  $2 \times 2$  matrices using 7 rather than 8 multiplications. To derive an algorithm that scales slower than the obvious  $O(N^3)$  cost of regular multiplication of  $N \times N$  matrices, consider the following recursive algorithm:

1. Take  $N \times N$  matrices with  $N = 2^k$ .
2. Consider block-matrix multiplication:

$$Q = AB = \begin{bmatrix} A_{0,0} & A_{0,1} \\ A_{1,0} & A_{1,1} \end{bmatrix},$$

where the blocks  $A_{i,j}$  and  $B_{i,j}$  are of size  $N/2 \times N/2$ .

3. Strassen's algorithm allows for the computation of the product  $Q$  as four submatrices  $Q_{i,j}$  of size  $N/2 \times N/2$ , using only 7 products of  $N/2 \times N/2$  submatrices.
4. The algorithm can be iterated on the 7 subproducts, until they are of size 1.
  - (i) Show that this algorithm requires  $\mu = N^{\log_2 7}$  multiplications, a substantial improvement over  $N^3$  since  $\log_2 7 = 2.80735$ .
  - (ii) Evaluate the number of additions of this recursive algorithm and compare to regular matrix multiplication.
  - (iii) Write out the algorithm in pseudocode.
  - (iv) Write a Matlab function for matrix multiplication of size  $2^K \times 2^K$  and compare running times for regular versus fast matrix multiplication. Comment on number of operations and running time.

*Solution:* TBD

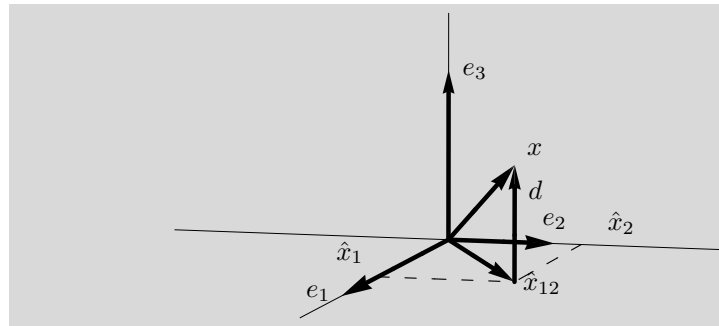
## Exercises

### 1.1. Multiplication by an Orthogonal Matrix

Prove that multiplication by an orthogonal matrix  $U$  preserves lengths (that is,  $\|Ux\| = \|x\|$  for any  $x$ ) and angles (that is  $\langle Ux, Uy \rangle = \langle x, y \rangle$  for any  $x$  and  $y$ ). Show also that the eigenvalues of  $U$  have unit absolute value.

### 1.2. Best Approximation in $\mathbb{R}^3$

Mimic what we did in  $\mathbb{R}^2$  in Section 1.1, and find the best approximation in  $\mathbb{R}^3$  of the vector  $x$  from Figure P1.2-1 in the  $(e_1, e_2)$ -plane. Find the difference between the vector and its approximation and prove it is the smallest possible.



**Figure P1.2-1:** Orthogonal projection onto a subspace. Here,  $x \in \mathbb{R}^3$  and  $\hat{x}_{12}$  is its projection onto the span of  $\{e_1, e_2\}$ . Note that  $x - \hat{x}_{12}$  is orthogonal to the span  $\{e_1, e_2\}$ , and that  $\hat{x}_{12}$  is the closest vector to  $x$  in the span of  $\{e_1, e_2\}$ .

### 1.3. Matrices Representing Bases and Frames

Given is the following matrix:

$$\Phi = [\varphi_0 \quad \varphi_1 \quad \varphi_2] = \begin{bmatrix} \sqrt{\frac{2}{3}} & -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{6}} \\ 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{bmatrix}.$$

- (i) Does this matrix represent a basis? If yes, is it an orthonormal basis? If no, why not?
- (ii) Each  $\varphi_i = [x_i \ y_i \ z_i]^T$ ,  $i = 0, 1, 2$  is a vector in a three-dimensional space  $\mathbb{R}^3$ . Project each  $\varphi_i$  onto the x-y plane, that is, the two-dimensional space  $\mathbb{R}^2$ . Write the resulting matrix  $\Phi'$ , where each vector is now in  $\mathbb{R}^2$ . Does this matrix represent a frame? If yes, is it a tight frame? If not, why not?

1.4. *Linear Independence*

Find the values of the parameter  $a \in \mathbb{C}$  such that the following set  $U$  is linearly independent

$$U = \left\{ \begin{bmatrix} 0 & a^2 \\ 0 & j \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 1 & a-1 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ ja & 1 \end{bmatrix} \right\}.$$

For  $a = j$ , express the matrix  $\begin{bmatrix} 0 & 5 \\ 2 & j-2 \end{bmatrix}$  as a linear combination of elements of  $U$ .

1.5. *Continuity of the Inner Product*

Show that an inner product is continuous in both of its variables, that is, that

$$\lim_{\|h_1\|, \|h_2\| \rightarrow 0} |\langle x + h_1, y + h_2 \rangle - \langle x, y \rangle| = 0.$$

1.6. *Inner Products on  $\mathbb{C}^N$* 

Prove that  $\langle x, y \rangle = y^* A x$  is a valid inner product if and only if  $A$  is a Hermitian, positive definite matrix.

1.7. *Norms on  $\mathbb{C}^N$* 

Consider the vector space  $\mathbb{C}^N$  of finite sequences  $x = [x_0 \ x_1 \ \cdots \ x_{N-1}]^T$ . Prove that  $v_1$  and  $v_2$  are norms on  $\mathbb{C}^N$  where

$$v_1(x) = \sum_{k=0}^{N-1} |x_k|, \quad v_2(x) = \left( \sum_{k=0}^{N-1} |x_k|^2 \right)^{1/2}. \quad (\text{P1.7-1})$$

(Hint: For  $v_2$ , use Minkowski's inequality (1.198a).)

1.8. *Orthogonal Transforms and  $\infty$  Norm*

Orthogonal transforms conserve the 2 norm, but not others, in general. Here we consider the  $\infty$  norm defined in (1.35b).

- (i) Consider the set of real orthogonal transforms on  $\mathbb{R}^2$ , that is, plane rotations and rotoinversions. Give the best lower and upper bounds  $a_2$  and  $b_2$  so that

$$a_2 \leq \|T_2 x\|_\infty \leq b_2$$

holds for all orthogonal  $T_2$  and all vectors  $x$  of unit 2 norm.

- (ii) Extend (i) and give the best lower and upper bounds  $a_N$  and  $b_N$  for the general case of  $\mathbb{R}^N$  with  $N \geq 2$ .

1.9. *Norms*

Let  $V$  be the set of all real-valued continuous functions defined on the interval  $[0, 1]$ , and define  $K_1$  and  $K_2$  as

$$K_1(x) = \int_0^1 |x(t)| dt, \quad K_2(x) = \left( \int_0^1 |x(t)|^2 dt \right)^{1/2}$$

- (i) Check that  $V$  is a vector space.
  - (ii) Prove that  $K_1$  and  $K_2$  are norms on  $V$ .
- (Hint: For  $K_2$ , use Minkowski's inequality (1.198b).)

1.10. *Cauchy-Schwarz Inequality, Triangle Inequality and the Parallelogram Law*

Prove the following:

- (i) Cauchy-Schwarz inequality given in (1.24).
- (ii) Triangle inequality given in Definition 1.9.
- (iii) Parallelogram law given in (1.25).

- (iv) A vector space  $V$  is an inner product space if and only if it is a normed vector space in which the parallelogram law holds. Use the *polarization identity*, which, for a real Hilbert space is

$$\langle x, y \rangle = \frac{1}{4} (\|x + y\|^2 - \|x - y\|^2), \quad (\text{P1.10-1a})$$

while, for a complex Hilbert space, it is

$$\langle x, y \rangle = \frac{1}{4} (\|x + y\|^2 - \|x - y\|^2 + j\|x + jy\|^2 - j\|x - jy\|^2). \quad (\text{P1.10-1b})$$

1.11. *Norm Induced by an Inner Product*

Let the mapping  $\langle \cdot, \cdot \rangle : V \times V \rightarrow \mathbb{C}$  be an inner product defined on the vector space  $V$ . Show that the function

$$n(x) = \sqrt{\langle x, x \rangle}, \quad \text{for } x \in V,$$

defines a norm on  $V$ .

1.12. *Convergence of the Inner Product in  $\ell^2(\mathbb{Z})$*

Let  $x$  and  $y$  be sequences in  $\ell^2(\mathbb{Z})$ . Show that the series (1.20b) defining  $\langle x, y \rangle$  converges absolutely. Since convergence of doubly-infinite series requires absolute convergence, this shows that the  $\ell^2(\mathbb{Z})$  inner product is always well defined for vectors in  $\ell^2(\mathbb{Z})$  (see Appendix 1.A.2).

1.13. *Distances Not Necessarily Induced By Norms*

A *distance*, or *metric*  $d : V \times V \rightarrow \mathbb{R}$  is a function with the following properties:

- (i) *Nonnegativity*:  $d(x, y) \geq 0$  for every  $x, y$  in  $V$ .
- (ii) *Symmetry*:  $d(x, y) = d(y, x)$  for every  $x, y$  in  $V$ .
- (iii) *Triangle inequality*:  $d(x, y) + d(y, z) \geq d(x, z)$  for every  $x, y, z$  in  $V$ .
- (iv) *Identity of Indiscernibles*:  $d(x, x) = 0$  and  $d(x, y) = 0$  implies  $x = y$ .

The *discrete metric* is given by

$$d(x, y) = \begin{cases} 0, & \text{if } x = y; \\ 1, & \text{if } x \neq y. \end{cases}$$

Show that the discrete metric is a valid distance and is not induced by any norm.

1.14. *Definition of  $\infty$ -norm*

Show that the  $\infty$  norm in (1.35b) is the natural extension of the  $p$  norm in (1.35a), by proving

$$\lim_{p \rightarrow \infty} \|x\|_p = \max_{i=0, 1, \dots, N-1} |x_i| \quad \text{for any } x \in \mathbb{R}^N.$$

(Hint: Normalize  $x$  by dividing by the entry of the largest magnitude. Compute the limit for the resulting vector.)

1.15. *Quasinorms with  $p < 1$*

Equation (1.35a) does not yield a valid norm when  $p < 1$ .

- (i) Show that Definition 1.9(iii) fails to hold for (1.35a) with  $p = 1/2$ .
- (ii) Show that for  $x \in \mathbb{R}^N$ ,  $\lim_{p \rightarrow 0} \|x\|_p^p$  gives the number of nonzero components in  $x$ .

1.16. *Equivalence of Norms on Finite-Dimensional Spaces*

Two norms  $\|\cdot\|_a$  and  $\|\cdot\|_b$  on a vector space  $V$  are called *equivalent* when there exist finite constants  $c_a$  and  $c_b$  such that

$$\|v\|_a \leq c_a \|v\|_b \quad \text{and} \quad \|v\|_b \leq c_b \|v\|_a \quad \text{for all } v \in V.$$

- (i) Show that the 1 norm, 2 norm, and  $\infty$  norm are equivalent by proving

$$\|x\|_\infty \leq \|x\|_2 \leq \|x\|_1$$

and

$$\|x\|_1 \leq \sqrt{N} \|x\|_2 \leq N \|x\|_\infty,$$

for all  $x \in \mathbb{R}^N$ .

- (ii) Show by counterexamples that the 1, 2 and  $\infty$  norms are not equivalent for infinite-dimensional spaces.

1.17. *Nesting of  $\ell^p$  Spaces*

Prove that if  $x \in \ell^p(\mathbb{Z})$  and  $p < q$ , then  $x \in \ell^q(\mathbb{Z})$ . This proves (1.37).

1.18.  *$\mathcal{L}^p$  Spaces*

- (i) Show that the parallelogram law holds in  $\mathcal{L}^2([0, 1])$ .  
 (ii) Show that the parallelogram law does not hold in  $\mathcal{L}^p([0, 1])$  for  $p \neq 2$ . To this end, consider the functions  $x(t) = t$  and  $y(t) = 1 - t$ .

This shows that, among all  $\mathcal{L}^p([0, 1])$  spaces, only  $\mathcal{L}^2([0, 1])$  is a candidate for being a Hilbert space.

1.19. *Closed Subspaces and  $\ell^0(\mathbb{Z})$*

Let  $\ell^0(\mathbb{Z})$  denote the set of complex-valued sequences with a finite number of nonzero entries.

- (i) Show that  $\ell^0(\mathbb{Z})$  is a subspace of  $\ell^2(\mathbb{Z})$ .  
 (ii) Show that  $\ell^0(\mathbb{Z})$  is not a closed subspace of  $\ell^2(\mathbb{Z})$ .

1.20. *Infinite Sequences and Completeness*

Consider the sequence

$$x = \left[ \dots \quad 0 \quad 0 \quad \boxed{1/\sqrt{2}} \quad 1/\sqrt{2} \quad 0 \quad 0 \quad \dots \right]^T,$$

where the boxed entry is at position zero. Is this sequence in  $\ell^2(\mathbb{Z})$ ? Why? If it is, does the set of sequences  $\{x_{n-2k}\}_{k \in \mathbb{Z}}$  form a basis for  $\ell^2(\mathbb{Z})$ ? Why?

1.21. *Completeness*

Consider the inner product space  $\mathcal{P}$  of all polynomials with

$$\langle p, q \rangle = \int_0^1 p(t)q^*(t) dt,$$

and let  $p_k$  be the following Cauchy sequence in  $\mathcal{P}$ :

$$p_k(t) = \sum_{i=0}^k \frac{1}{2^i} t^i.$$

Prove that  $\mathcal{P} \subset \mathcal{L}^2([0, 1])$  is not a Hilbert space.

1.22. *Completeness of  $\mathbb{C}^N$*

Prove that  $\mathbb{C}^N$  equipped with the  $p$  norm is a complete vector space for any  $p \in [1, \infty)$  or  $p = \infty$ . You may assume that  $\mathbb{C}$  itself is complete.

(Hint: Show that having a Cauchy sequence in  $\mathbb{C}^N$  implies each of the  $N$  components is a Cauchy sequence.)

1.23. *Cauchy Sequences*

Show that in a normed vector space, every convergent sequence is a Cauchy sequence.

1.24. *Norms of Operators*

- (i) Consider a symmetric matrix  $A$  with

$$A = \begin{bmatrix} 1 & 3 \\ 3 & 1 \end{bmatrix}.$$

Calculate  $\|A\|$  and  $\|A^{-1}\|$ .

(Hint: Calculate the eigenvalues of the matrix  $A$ .)

- (ii) Consider operators that map  $\ell^2(\mathbb{Z})$  to itself. For the following operators, indicate their norm, or bounds on their norm.

(i)

$$(Ax)_n = m_n \cdot x_n, \quad m_n = e^{j\Theta_n}, \quad n \in \mathbb{Z}.$$



(ii)

$$(Ax)_{2n} = x_{2n} + x_{2n+1}, \quad (Ax)_{2n+1} = x_{2n} - x_{2n+1}, \quad n \in \mathbb{Z}.$$

1.25. *Relation Between Operator Norm and Eigenvalues*Show that for any  $A \in \mathbb{C}^{M \times N}$ ,

$$\|A\|_2 = \sqrt{\lambda_{\max}(A^*A)},$$

where  $\lambda_{\max}$  denotes the largest eigenvalue.*Hint:* Apply the diagonalization (1.223a) to  $A^*A$ .1.26. *Adjoint Operators*

Prove parts (iv), (vi), (vii), and (viii) of Theorem 1.21.

1.27. *Eigenvalues of Definite Operators*Let  $A$  be a self-adjoint operator on Hilbert space  $H$ .

- (i) Let  $\lambda$  be an eigenvalue of  $A$ . Show that  $A$  positive semidefinite implies  $\lambda \geq 0$  and furthermore  $A$  positive definite implies  $\lambda > 0$ .
- (ii) Show that the existence of a nonpositive eigenvalue implies  $A$  is not positive definite and furthermore the existence of a negative eigenvalue implies  $A$  is not positive semidefinite.

1.28. *Operator Expansion*Let  $A : H \rightarrow H$  be a bounded linear operator with  $\|A\| < 1$ .

- (i) Show that  $I - A$  is invertible.
- (ii) Show that, for every  $y$  in  $H$ ,

$$(I - A)^{-1}y = \sum_{k=0}^{\infty} A^k y.$$

- (iii) In practice one can only compute a finite number of terms in the series. For  $\|y\| = 1$  and  $K$  terms in the expansion, find an upper-bound on the error:

$$\left\| (I - A)^{-1}y - \sum_{k=0}^{K-1} A^k y \right\|.$$

1.29. *Projection Operators*

Let

$$B = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}.$$

Find all projection operators onto the range of  $B$ . Specify which one is the orthogonal projection operator onto the range of  $B$ .1.30. *Projection via Domain Restriction*Recall the definition of  $1_{\mathcal{I}} : \mathcal{L}^2(\mathbb{R}) \rightarrow \mathcal{L}^2(\mathbb{R})$  in (1.58).

- (i) Show that  $1_{\mathcal{I}}$  is an orthogonal projection operator.
- (ii) Show that if  $\mathcal{I}_1$  and  $\mathcal{I}_2$  are disjoint subsets of  $\mathbb{R}$ , the ranges of the associated operators,  $\mathcal{R}(1_{\mathcal{I}_1})$  and  $\mathcal{R}(1_{\mathcal{I}_2})$ , are orthogonal.
- (iii) Under what condition does the orthogonal decomposition  $\mathcal{L}^2(\mathbb{R}) = \mathcal{R}(1_{\mathcal{I}_1}) \oplus \mathcal{R}(1_{\mathcal{I}_2})$  hold?

1.31. *Inverses, Adjoint, and Projection Operators*

Prove Theorem 1.29.

1.32. *Systems of Linearly Independent Vectors*Given a set of  $M$ -dimensional linearly independent vectors  $\{a_0, a_1, \dots, a_{N-1}\}$ , with  $N < M$ , and a vector  $b$  in  $\mathbb{R}^M$  outside their span:

- (i) Is the vector set  $\{a_0, a_1, \dots, a_{N-1}, b\}$  a basis? Explain.
- (ii) Give an expression for the distance between  $b$  and the projection of  $x \in \mathbb{R}^N$  onto  $A = \{a_0, a_1, \dots, a_{N-1}\}$ .

- (iii) Write the equations for the components of the error vector.
- (iv) Find the least-squares solution  $\hat{x}$  and show that the projection of  $b$  onto the space spanned by the columns of  $A$  can be computed with the use of  $P = A(A^T A)^{-1} A^T$ .
- (v) Prove that  $P$  is an orthogonal projection matrix.

1.33. *An Inner Product on Random Vectors*

Consider the set of complex random vectors of length  $N$  with the restriction that each component has finite variance. This set forms a vector space over  $\mathbb{C}$ .

- (i) Show that  $\langle x, y \rangle = \sum_{k=0}^{N-1} E[x_k y_k^*]$  is a valid inner product on this set. In doing so, explain the meaning of  $x = 0$ .
- (ii) What condition makes random vectors orthogonal under this inner product?
- (iii) Zero-mean random vectors  $x$  and  $y$  are called uncorrelated when the matrix  $E[xy^T]$  is all 0s. Are orthogonal vectors uncorrelated? Are uncorrelated vectors orthogonal?
- (iv) Let  $x$  and  $y$  be zero-mean Gaussian vectors. Does  $\langle x, y \rangle = 0$  imply  $x$  and  $y$  are independent? If not, what weaker condition is implied?

Note that the inner product defined in this exercise is not often useful in deriving optimal estimators; see Solved Exercise 1.34.

1.34. *Bayesian Linear MMSE Estimation via Orthogonality Principle*

As in Example 1.64, let  $x$  and  $y$  be jointly-distributed real random vectors with  $E[x] = \mu_X$ ,  $E[y] = \mu_Y$ , and

$$E \begin{bmatrix} x \\ y \end{bmatrix} \begin{bmatrix} x^T & y^T \end{bmatrix} = \begin{bmatrix} \Sigma_x & \Sigma_{xy} \\ \Sigma_{xy}^T & \Sigma_y \end{bmatrix}.$$

Use the projection theorem to find the linear minimum mean-squared error (LMMSE) estimator of  $x$  as a function of  $y$ , that is, the optimal estimator of the form  $\hat{x} = Ay + b$ .

1.35. *Riesz Bases*

- (i) Prove that the standard basis in  $\ell^2(\mathbb{Z})$  is a Riesz basis with  $\lambda_{\min} = \lambda_{\max} = 1$ .
- (ii) Let  $\{e_k\}_{k \in \mathbb{Z}}$  denote the standard basis in  $\ell^2(\mathbb{Z})$  and define the following scaled version:

$$\varphi_k = 2^k e_k, \quad k \in \mathbb{Z}.$$

Prove that  $\{\varphi_k\}_{k \in \mathbb{Z}}$  is a basis, but there is neither a positive  $\lambda_{\min}$  nor a finite  $\lambda_{\max}$  such that (1.80) in the definition of Riesz basis holds.

- (iii) Let

$$\psi_k = \cos(k) e_k, \quad k \in \mathbb{Z}.$$

Prove or disprove that  $\{\psi_k\}_{k \in \mathbb{Z}}$  is a basis for  $\ell^2(\mathbb{Z})$ , and prove or disprove that  $\{\psi_k\}_{k \in \mathbb{Z}}$  is a Riesz basis for  $\ell^2(\mathbb{Z})$ .

1.36. *Basis that is not a Riesz Basis*

Complete Example 1.29(ii) by showing that

$$\varphi_k = \sum_{i=0}^k (k+1)^{-1/2} e_i, \quad k \in \mathbb{N}.$$

is a basis for  $\ell^2(\mathbb{N})$  but not a Riesz basis.

1.37.  *$\ell^p$  Norms in Different Bases*

Consider  $\mathbb{R}^2$  and the standard basis  $\{e_0, e_1\}$  as well as another orthonormal basis  $\{\varphi_0, \varphi_1\}$ . Take any vector  $x$  expressed as  $\begin{bmatrix} x_0 & x_1 \end{bmatrix}^T$  in the standard basis, and as  $\begin{bmatrix} \alpha_0 & \alpha_1 \end{bmatrix}^T$  in the  $\{\varphi_0, \varphi_1\}$  basis with  $\alpha_k = \langle x, \varphi_k \rangle$ , for  $k = 0, 1$ .

- (i) Among  $\ell^p$  norms, for  $p \in [1, \infty]$ , show that the 2 norm is the only one invariant with respect to the basis, that is, for arbitrary vectors,

$$\left\| \begin{bmatrix} x_0 & x_1 \end{bmatrix}^T \right\|_p = \left\| \begin{bmatrix} \alpha_0 & \alpha_1 \end{bmatrix}^T \right\|_p,$$

hold only for  $p = 2$ .

- (ii) Generalize this to biorthogonal bases in  $\mathbb{R}^2$ .

(iii) What can you say about arbitrary dimensions?

1.38. *Symmetric and Antisymmetric Functions*

Consider the vector space of real functions in  $\mathcal{L}^2([-\pi, \pi])$  over the real numbers, i.e., square-integrable real functions on the interval  $[-\pi, \pi]$ . It has an orthonormal basis given by

$$\left\{ \frac{1}{\sqrt{2\pi}}, \frac{\cos t}{\sqrt{\pi}}, \frac{\sin t}{\sqrt{\pi}}, \frac{\cos 2t}{\sqrt{\pi}}, \frac{\sin 2t}{\sqrt{\pi}}, \frac{\cos 3t}{\sqrt{\pi}}, \frac{\sin 3t}{\sqrt{\pi}}, \dots \right\}.$$

Consider the following two subspaces:  $S$  – space of symmetric functions, that is,  $f(t) = f(-t)$ , and  $A$  – space of antisymmetric functions,  $f(t) = -f(-t)$ .

- (i) Show how any function  $f(t)$  from  $\mathcal{L}^2([-\pi, \pi])$  can be written as  $f(t) = f_s(t) + f_a(t)$  with  $f_s(t) \in S$  and  $f_a(t) \in A$ .
- (ii) Give orthonormal bases for  $S$  and  $A$ .
- (iii) Verify that  $\mathcal{L}^2([-\pi, \pi]) = S \oplus A$ .

1.39. *Orthonormal Sets*

Let  $E = \mathcal{L}^2([-\pi, \pi])$ .

- (i) Consider the set  $\phi_k(t) = \frac{1}{\sqrt{2\pi}} \exp(jkt)$  for  $k \in \mathbb{Z}$ . Prove that  $\{\phi_k(t)\}_{k \in \mathbb{Z}}$  is an orthonormal set in  $E$ .
- (ii) Consider  $\{\varphi_k(t) = \frac{1}{\sqrt{\pi}} \sin(kt)\}_{k \geq 1}$ . Prove that  $\{\varphi_k(t)\}_{k \geq 1}$  is an orthonormal set in  $E$  but that it is not a basis.
- (iii) Consider  $\{\psi_k(t) = \frac{1}{\sqrt{\pi}} \cos(kt)\}_{k \geq 1}$ . Show that the set  $\{\frac{1}{\sqrt{2\pi}}, \varphi_k(t), \psi_k(t)\}_{k \geq 1}$  is an orthonormal set in  $E$ .

(Hint: See also Problem 1.38.)

1.40. *Least-Squares Approximation in an Orthonormal Representation*

Assume a finite-dimensional space  $\mathbb{R}^N$  and an orthonormal basis  $\{\varphi_1, \varphi_2, \dots, \varphi_N\}$ . Any vector  $x$  can thus be written as  $x = \sum_{i=1}^N \alpha_i \varphi_i$  where  $\alpha_i = \langle x, \varphi_i \rangle$ . Consider the best approximation to  $x$  in the least-squares sense and living on the subspace spanned by the first  $k$  vectors,  $\{\varphi_1, \varphi_2, \dots, \varphi_k\}$ , or  $\hat{x} = \sum_{i=1}^k \beta_i \varphi_i$ . Prove that  $\beta_i = \alpha_i$  for  $i = 1, 2, \dots, k$ , by showing that it minimizes  $\|x - \hat{x}\|$ .

(Hint: Use Parseval's equality.)

1.41. *Orthogonal Projection in  $\mathbb{R}^N$*

Let  $\{\varphi_k\}_{k \in \mathcal{K}}$  be an orthonormal set. Prove that

$$Px = \sum_{k \in \mathcal{K}} \langle x, \varphi_k \rangle \varphi_k$$

is an orthogonal projection operator onto  $\overline{\text{span}}(\{\varphi_k\}_{k \in \mathcal{K}})$ .

1.42. *Generalized Parseval's Equalities*

(1.109) and (1.110) using the biorthogonality of the pair of bases.

1.43. *Biorthogonal Pair of Bases of Cosine Functions*

Consider the set  $\Phi = \{\varphi_k\}_{k \in \mathbb{N}}$  defined in Example 1.31 and the sets  $\Psi = \{\psi_k\}_{k \in \mathbb{N}}$  and  $\tilde{\Psi} = \{\tilde{\psi}_k\}_{k \in \mathbb{N}}$  defined in Example 1.37.

- (i) Show that  $\Psi$  and  $\tilde{\Psi}$  satisfy the biorthogonality condition (1.102).
- (ii) Show that  $\overline{\text{span}}(\Psi) = \overline{\text{span}}(\tilde{\Psi}) = \overline{\text{span}}(\Phi)$ .

1.44. *Dual Bases*

Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{K}}$  be a Riesz basis for Hilbert space  $H$  with constants  $\lambda_{\min}$  and  $\lambda_{\max}$ .

- (i) Show that the dual of the dual of  $\Phi$  is  $\Phi$ .
- (ii) Show that the dual of  $\Phi$  is  $\Phi$  if and only if  $\Phi$  is an orthonormal basis.
- (iii) Show that the dual of  $\Phi$  is a Riesz basis with constants  $1/\lambda_{\max}$  and  $1/\lambda_{\min}$ .

1.45. *Oblique Projection Property*

Prove Theorem 1.46.

1.46. *Normal Equations*

Verify that  $\hat{x}$  in (1.127) is the orthogonal projection of  $x$  onto the subspace  $\overline{\text{span}}(\{\psi_k\}_{k \in \mathcal{I}})$ . Consider both the case when  $\{\psi_k\}_{k \in \mathcal{I}}$  are linearly independent as well as when they are linearly dependent.

1.47. *Orthogonal Projection in Coefficient Space*

Let  $\{\varphi_k\}_{k \in \mathcal{K}}$  be a basis for the Hilbert space  $H$ , and let  $\{\psi_k\}_{k \in \mathcal{I}}$  be another set in  $H$ . Denote the orthogonal projection of  $x$  onto  $\overline{\text{span}}(\{\psi_k\}_{k \in \mathcal{I}})$  by  $\hat{x}$ , and denote representations of  $x$  and  $\hat{x}$  with respect to  $\{\varphi_k\}_{k \in \mathcal{K}}$  by  $x = \Phi\alpha$  and  $\hat{x} = \Phi\hat{\alpha}$ .

- (i) Find an expression for  $P : \ell^2(\mathcal{K}) \rightarrow \ell^2(\mathcal{K})$  that maps  $\alpha$  to  $\hat{\alpha}$ . You may use the operators  $\Phi$ ,  $\tilde{\Phi}$ , and  $\Psi$ .
- (ii) Show that  $P$  is a projection operator.
- (iii) Show that  $P$  is an orthogonal projection operator if and only if  $\{\varphi_k\}_{k \in \mathcal{K}}$  is an orthonormal basis for  $H$ .

1.48. *Successive Approximation with Nonorthogonal Basis*

Let  $\{\varphi_i\}_{i \in \mathbb{N}}$  be a linearly independent set in the Hilbert space  $H$ . For each  $k \in \mathbb{N}$ , let  $S_k = \text{span}(\{\varphi_0, \varphi_1, \dots, \varphi_{k-1}\})$  and  $\hat{x}^{(k)}$  denote the best approximation of  $x$  in  $S_k$ . Prove that the recursive algorithm given in (1.131) provides a sequence of best approximations that all satisfy the normal equations (1.128a).

(Hint: Use induction over  $k$ .)

1.49. *Exploring the Definition of Frame*

Let  $\Psi = \{\psi_k\}_{k \in \mathcal{J}} \subset H$  be a frame.

- (i) Show that if  $\Psi$  is not linearly independent, the following is *not* true: For any expansion  $x = \sum_{k \in \mathcal{J}} \alpha_k \psi_k$ , condition (1.80) holds.
- (ii) Show that for any  $x \in H$ , there exists an expansion  $x = \sum_{k \in \mathcal{J}} \alpha_k \psi_k$  such that condition (1.80) holds.

1.50. *Frame of Cosine Functions*

Find the lower and upper frame bounds for the frame

$$\{1, \sqrt{2} \cos(\pi t)\} \cup \{\sqrt{2} \cos(2\pi kt), 2 \cos(\pi t) \cos(2\pi kt)\}_{k \in \mathbb{Z}^+}$$

given in Example 1.44.

1.51. *Dual Frame*

Consider the space  $\mathbb{R}^4$ . It is clear that

$$\varphi_{k,n} = \delta_{n-k}$$

for  $k = 0, 1, \dots, 3$ ,  $n = 0, 1, \dots, 3$ , form a basis for this space.

- (i) Consider

$$\psi_{k,n} = \delta_{n-k} - \delta_{n-k-1},$$

for  $k = 0, 1, 2$ , and  $\psi_{3,n} = -\delta_{n-1} + \delta_{n-4}$ . Show that the 4-dimensional set  $\{\psi_k\}_{k=0}^3$  does not form a basis for  $\mathbb{R}^4$ . Which vectors in  $\mathbb{R}^4$  are not in  $\text{span}(\{\psi_k\}_{k=0}^3)$ ?

- (ii) Show that  $F = \{\varphi_k, \psi_k\}_{k=0}^3$  is a frame. Compute the frame bounds  $A$  and  $B$ .
- (iii) Find the canonical dual frame to  $F$ .
- (iv) Let the vectors  $\varphi_k$  and  $\psi_k$  be given as columns of the following two matrices:

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}.$$

Comment on the completeness of  $\Phi$  and  $\Psi$  in  $\mathbb{R}^4$ , show that  $F = [\Phi \ \Psi]$  is a frame, and find its frame bounds as well as a dual frame.

1.52. *Properties of Dual Pair of Frames*

This exercise develops results for dual pairs of frames that parallel the results for biorthogonal pairs of bases established in Section 1.5.3. Let  $\Phi = \{\varphi_k\}_{k \in \mathcal{J}}$  and  $\tilde{\Phi} = \{\tilde{\varphi}_k\}_{k \in \mathcal{J}}$  be a dual pair of frames for Hilbert space  $H$ .

- (i) Show that (1.144) holding for every  $x$  in  $H$  implies (1.145) holds for every  $x$  in  $H$ . This establishes that the roles of the two frames in a dual pair can be reversed.
- (ii) Let  $x$  and  $y$  be any vectors in  $H$ . Show that if  $\tilde{\alpha} = \Phi^*x$  and  $\beta = \tilde{\Phi}^*y$ , then  $\langle x, y \rangle = \langle \tilde{\alpha}, \beta \rangle$ . This shows how frame expansions enable computation of an inner product in  $H$  through an  $\ell^2(\mathcal{J})$  inner product.
- (iii) Let  $\mathcal{I} \subseteq \mathcal{J}$  and recall the notations (1.94) and (1.95) for restricted synthesis and analysis operators. Determine a sufficient condition under which  $\Phi_{\mathcal{I}}\tilde{\Phi}_{\mathcal{I}}^*$  is a projection operator. (Forcing  $\Phi$  and  $\tilde{\Phi}$  to form a biorthogonal pair of bases is not an adequate answer.)

1.53. *Random Frame*

Let  $W_N = e^{-j2\pi/N}$ . Consider the square  $[-1, 1] \times [-1, 1]$  of pairs in  $\mathbb{R}^2$ . Pick  $N$  points at random with uniform distribution from this set, let their coordinates be  $(x_i, y_i)$ ,  $i = 1, 2, \dots, N$ . Create the matrix  $M$  having the entries  $(x_i, y_i)$  as rows. What are the condition(s) that the pairs  $(x_i, y_i)$  should fulfill so that  $M$  is a frame?  
(Hint: This is a purely geometrical exercise. Think of what is needed to represent a vector in a two-dimensional space.)

1.54. *Tight Frames as Projections from Orthonormal Bases*

Let  $W_N = e^{-j2\pi/N}$ .

- (i) Consider a frame expansion of  $\mathbb{R}^2$  that is given by the  $N$  vectors  $\phi_k$ , with

$$\phi_k = [\Re\{W_N^k\} \quad \Im\{W_N^k\}]^T, \quad k = 0, \dots, N-1. \quad (\text{P1.54-1})$$

Plot the vectors of the frame on the unit circle, for  $N = 3, 4, 5$ .

- (ii) Consider now the DFT matrix:

$$F = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_N & W_N^2 & \dots & W_N^{N-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \dots & W_N^{(N-1)(N-1)} \end{bmatrix}$$

and let  $F_p$  be the matrix formed by its first  $p$  columns and normalized correspondingly; that is, having as its  $i$ th column vector (for  $i = 0, \dots, p-1$ )

$$f_{pi} = \frac{1}{\sqrt{p}} [W_N^{ni}]_n, \quad n = 0, \dots, N-1.$$

Show that  $F_p$  is a tight frame with redundancy factor  $N/p$ , that is

$$F_p^* F_p = \frac{N}{p} I_p.$$

(Hint: Use the identity  $\sum_{n=0}^{N-1} W_N^{n\ell} = N\delta_{\ell-N}$ , where  $\delta_n$  is Kronecker delta function, and  $\ell \in \mathbb{Z}$ .)

1.55. *Tight Frame with Nonequal-Norm Vectors*

Assume  $\alpha \in \mathbb{R}$ ,  $\alpha \neq 0$ , and take the following set of vectors:

$$\varphi_0 = \begin{bmatrix} 0 \\ \alpha \end{bmatrix}, \quad \varphi_1 = \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix}, \quad \varphi_2 = \begin{bmatrix} -\cos \theta \\ \sin \theta \end{bmatrix}.$$

For which values of  $\theta$  and  $\alpha$  is the above set a tight frame? Draw a few representative cases.

1.56. *Tight Frame of Affine Functions*

In  $\mathcal{L}^2([0, 1])$ , consider the subspace  $S$  of affine functions. Find a 4-element tight frame  $\Phi$  for  $S$  that includes  $\varphi_0(t) = 1$  and  $\varphi_1(t) = \sqrt{3}t$ . Also find the frame bound  $\lambda$  and the canonical dual frame.

Hint: A union of orthonormal bases is always a tight frame.

1.57. *Relation Between Bases*

Given are two different bases  $\Phi = \{\varphi_k\}_{k \in \mathbb{Z}}$ ,  $\Psi = \{\psi_k\}_{k \in \mathbb{Z}}$ , for a Hilbert space  $H$ . Show how to express one in terms of the other.

1.58. *Change of Basis*

Given are two different bases  $\Phi = \{\varphi_k\}_{k \in \mathbb{Z}}$ ,  $\Psi = \{\psi_k\}_{k \in \mathbb{Z}}$ , for a Hilbert space  $H$ . Show how to change the representation of a given  $x \in H$  from one basis to the other, that is,

$$x = \sum_{k \in \mathcal{K}} \alpha_k \varphi_k = \sum_{k \in \mathcal{K}} \beta_k \psi_k.$$

Calculate it when  $\Phi$  and  $\Psi$  are

$$\Phi = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad \Psi = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}.$$

1.59. *Complex Multiplication*

Multiplying two complex numbers

$$e + jf = (a + jb)(c + jd),$$

can be written as

$$\begin{bmatrix} e \\ f \end{bmatrix} = \begin{bmatrix} c & -d \\ d & c \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix},$$

which takes 4 multiplications and 2 additions. Show that this can be computed with 3 multiplications and 5 additions.

1.60. *Gaussian Elimination*

Our aim is to solve the system of linear equations  $Ax = y$  (general conditions for existence of a solution are given in Appendix 1.B.1). Comment on whether the solution to each of the following systems of equations exists, and if it does, find it.

(i)

$$A = \begin{bmatrix} 1 & 0 & 3 \\ 4 & 5 & 2 \\ -1 & -1 & 2 \end{bmatrix}, \quad y = \begin{bmatrix} 10 \\ 20 \\ 3 \end{bmatrix}.$$

(ii)

$$A = \begin{bmatrix} 1 & 0 & 2 \\ 4 & 5 & 8 \\ -1 & -1 & -2 \end{bmatrix}, \quad y = \begin{bmatrix} 7 \\ 38 \\ -9 \end{bmatrix}.$$

(iii)

$$A = \begin{bmatrix} 1 & 0 & 2 \\ 4 & 5 & 8 \\ -1 & -1 & -2 \end{bmatrix}, \quad y = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}.$$

1.61. *Kaczmarz's Algorithm*

Consider Example 1.63 and prove the following facts when the size- $N \times N$  matrix  $A$  is of rank  $N$ , and thus the set of rows  $r_n$ ,  $n = 0, 1, \dots, N-1$  is linearly independent.

- (i) Show that the intersection of the affine subspaces  $S_n$ ,  $n = 0, 1, \dots, N-1$ , is unique and specifies the solution of the linear system  $Ax = b$ .
- (ii) Prove that when the rows are mutually orthogonal, Kaczmarz's algorithm converges in  $N$  steps or only one sweep.

1.62. *Sums and Products*

Compute the following sums and products:

- (i)  $\prod_{k=1}^{\infty} \exp(\frac{j2\pi}{k(k+1)})$
- (ii)  $\sum_{k=1}^{1024} \exp(\frac{j2\pi k}{16})$
- (iii)  $\sum_{k=0}^{\infty} a^k z^{-k}$  (specify also the region of convergence, meaning the values of  $z$  for which there is convergence)

1.63. *Convergence of Sequences*

Let  $\{a_k\}_{k=0}^{\infty}$  converge to  $a$  and  $\{b_k\}_{k=0}^{\infty}$  converge to  $b$ . Prove the following:

- (i) If  $c$  is some real number,  $ca_0, ca_1, \dots$  converges to  $ca$ .
- (ii)  $a_0 + b_0, a_1 + b_1, \dots$  converges to  $a + b$ .

- (iii)  $a_0b_0, a_1b_1, \dots$  converges to  $ab$ .  
 (iv) If  $a_k \neq 0$  for each  $k \in \mathbb{N}$  and  $a \neq 0$ ,  $b_0/a_0, b_1/a_1, \dots$  converges to  $b/a$ .

f

1.64. *Convergence Tests*

Let  $\{a_k\}_{k=1}^{\infty}$  and  $\{b_k\}_{k=1}^{\infty}$  be sequences that satisfy  $0 \leq a_k \leq b_k$  for every  $k \in \mathbb{Z}^+$ . Then

- if  $\sum_{k=1}^{\infty} b_k$  converges,  $\sum_{k=1}^{\infty} a_k$  converges as well; and
- if  $\sum_{k=1}^{\infty} a_k$  diverges,  $\sum_{k=1}^{\infty} b_k$  diverges as well.

The *ratio test* for convergence of  $\sum_{k=1}^{\infty} a_k$  states that, given

$$\lim_{k \rightarrow \infty} \left| \frac{a_{k+1}}{a_k} \right| = L, \quad (\text{P1.64-1})$$

- if  $0 \leq L < 1$ , the series converges absolutely;
- if  $1 < L$  or  $L = \infty$ , the series diverges; and
- $L = 1$ , the test is inconclusive, the series could converge or diverge. The convergence needs to be determined another way.

Based on the above, determine whether  $\sum_{k=1}^{\infty} c_k$  converges for each of the following sequences:

- (i)  $c_k = k^2/(2k^4 - 3)$ .  
 (ii)  $c_k = \log k/k$ .  
 (iii)  $c_k = k^k/k!$ .  
 (iv)  $c_k = a^k/k!$ .

1.65. *Useful Series*

In this exercise, we explore a few useful series.

- (i) *Finite Sum*: Prove the following formula:

$$\sum_{k=0}^{N-1} t^k = \frac{1-t^N}{1-t}. \quad (\text{P1.65-1})$$

- (ii) *Geometric Series*: Determine conditions on when

$$\sum_{k=0}^{\infty} t^k \quad (\text{P1.65-2})$$

converges. Prove that when it converges its sum is given by

$$\frac{1}{1-t}. \quad (\text{P1.65-3})$$

- (iii) *Power Series*: Determine whether

$$\sum_{k=1}^{\infty} a_k t^k \quad (\text{P1.65-4})$$

converges, as well as when and how.

- (iv) *Taylor Series*: If a function  $x(t)$  has  $(n+1)$  continuous derivatives, then it can be expanded into a *Taylor series* around a point  $t_0$  as follows:

$$x(t) = \sum_{k=0}^n \frac{(t-t_0)^k}{k!} x^{(k)}(t_0) + R_n, \quad R_n = \frac{(t-t_0)^{(n+1)}}{(n+1)!} x^{(n+1)}(\xi) \quad (\text{P1.65-5})$$

for some  $\xi$  between  $t$  and  $t_0$ . Find the Taylor series expansion of  $x(t) = 1/(1-t)$  around a point  $t_0$ .

Function	Expansion	Function	Expansion
$\sin t$	$\sum_{k=0}^{\infty} (-1)^k \frac{t^{(2k+1)}}{(2k+1)!}$	$\cos t$	$\sum_{k=0}^{\infty} (-1)^k \frac{t^{(2k)}}{(2k)!}$
$e^t$	$\sum_{k=0}^{\infty} \frac{t^k}{(k)!}$	$a^t$	$\sum_{k=0}^{\infty} \frac{(t \ln a)^k}{(k)!}$
$\sinh t$	$\sum_{k=0}^{\infty} \frac{t^{(2k+1)}}{(2k+1)!}$	$\cosh t$	$\sum_{k=0}^{\infty} \frac{t^{(2k)}}{(2k)!}$
$\ln(1+t)$	$\sum_{k=1}^{\infty} (-1)^{(k+1)} \frac{t^k}{k}$	$\ln \frac{1+t}{1-t}$	$\sum_{k=1}^{\infty} 2 \frac{t^{(2k-1)}}{2k-1}$

**Table P1.65-1:** Useful MacLaurin series expansions.

(v) *MacLaurin Series:* For  $a = 0$ , the Taylor series is called the *MacLaurin series*:

$$f(t) = \sum_{k=0}^n \frac{t^k}{k!} f^{(k)}(0) + R_n. \quad (\text{P1.65-6})$$

Find the MacLaurin series expansion of  $f(t) = 1/(1-t)$ .

1.66. *Eigenvalues and Eigenvectors*

Consider the matrices  $A = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}$  and  $B = \begin{bmatrix} \alpha & \beta \\ \beta & \alpha \end{bmatrix}$  where  $\alpha$  and  $\beta$  cannot be equal to zero at the same time.

- Give the eigenvalues and eigenvectors for  $A$  and  $B$  (make sure that the eigenvectors are of norm 1).
- Show that  $A = VDV^T$  where the columns of  $V$  correspond to the eigenvectors of  $A$  and  $D$  is a diagonal matrix whose main diagonal corresponds to the eigenvalues of  $A$ . Is  $V$  a unitary matrix? Why?
- Check your results using the built-in Matlab function `eig`.
- Compute the determinants of  $A$  and  $B$ . Is  $A$  invertible? If it is, give its inverse; if not, say why.
- When is the matrix  $B$  invertible? Compute  $B^{-1}$  when it exists.

1.67. *Operator Norm, Singular Values and Eigenvalues*

For matrix  $A$  (bounded linear operator), show the following:

- If the matrix  $A$  is Hermitian, for every nonnegative eigenvalue  $\lambda_k$ , there is an identical singular value  $\sigma_i = \lambda_k$ . For every negative eigenvalue  $\lambda_k$ , there is a corresponding singular value  $\sigma_i = |\lambda_k|$ .
- If the matrix  $A$  is Hermitian with eigenvalues  $\lambda_k$ ,

$$\|A\|_2 = \max\{\lambda_k\}.$$

- Call  $\mu_k$  the eigenvalues of  $A^*A$ ; then

$$\|A\|_2 = \sqrt{\max\{\mu_k\}} = \max\{\sigma_k\}.$$

1.68. *Least-Squares Solution to a Linear System of Equations*

The general solution to this problem was given in (1.208).

- Show that if  $y$  belongs to the column space of  $A$ , then  $\hat{y} = y$ .
- Show that if  $y$  is orthogonal to the column space of  $A$ , then  $\hat{y} = 0$ .



- (iii) Show that for the least-squares solution, the partial derivatives  $\partial(|y - \hat{y}|^2)/\partial \hat{x}_i$  are all zero.

1.69. *Power of a Matrix*

Given a square, invertible matrix  $A$ , find an expression for  $A^k$  as a function of the eigenvectors and eigenvalues of  $A$ .

1.70. *Conditional Distributions of Jointly Gaussian Vectors*

Let  $\begin{bmatrix} y^T & z^T \end{bmatrix}^T$  be a jointly Gaussian vector with

$$\mathbb{E} \begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} \mu_y \\ \mu_z \end{bmatrix}, \quad \text{and} \quad \mathbb{E} \begin{bmatrix} y \\ z \end{bmatrix} \begin{bmatrix} y^T & z^T \end{bmatrix} = \begin{bmatrix} \Sigma_y & \Sigma_{yz} \\ \Sigma_{zy} & \Sigma_z \end{bmatrix}.$$

- (i) Let  $A$  be a matrix with dimensions such that  $Ay$  is defined. Show that  $Ay$  is a jointly Gaussian vector with mean  $A\mu_y$  and covariance matrix  $A\Sigma_y A^T$ .
- (ii) Why must we have  $\Sigma_y = \Sigma_y^T$ ,  $\Sigma_z = \Sigma_z^T$ , and  $\Sigma_{yz} = \Sigma_{zy}^T$ ?
- (iii) Using the joint PDF of  $\begin{bmatrix} y^T & z^T \end{bmatrix}^T$ , find the joint PDF of  $y$ , thus showing that  $y$  is jointly Gaussian with mean  $\mu_y$  and covariance  $\Sigma_y$ .
- (iv) Using the joint PDF of  $\begin{bmatrix} y^T & z^T \end{bmatrix}^T$  and the joint PDF of  $y$ , show that the conditional distribution of  $y$  given  $z = t$  is jointly Gaussian with mean  $\mu_y + \Sigma_{yz}\Sigma_z^{-1}(t - \mu_z)$  and covariance  $\Sigma_y - \Sigma_{yz}\Sigma_z^{-1}\Sigma_{zy}$ .



## Chapter 2

# Sequences and Discrete-Time Systems

## Contents

2.1	Introduction . . . . .	172
2.2	Sequences . . . . .	175
2.3	Systems . . . . .	185
2.4	Discrete-Time Fourier Transform . . . . .	205
2.5	$z$ -Transform . . . . .	223
2.6	Discrete Fourier Transform . . . . .	240
2.7	Multirate Sequences and Systems . . . . .	248
2.8	Discrete Stochastic Processes and Systems . . . . .	268
2.9	Computational Aspects . . . . .	281
2.A	Elements of Analysis . . . . .	290
2.B	Elements of Algebra . . . . .	293
	Chapter at a Glance . . . . .	300
	Historical Remarks . . . . .	303
	Further Reading . . . . .	303
	Exercises with Solutions . . . . .	304
	Exercises . . . . .	310

Time is ordered—from past to future. Any countably-infinite set of times can be indexed by the integers to maintain this order, and associating the integers with *discrete time* prompts us to refer to doubly-infinite sequences as *discrete-time signals*. As we saw in the previous chapter, these sequences form the vector space  $\mathbb{C}^{\mathbb{Z}}$  (assuming they are complex-valued). Operators that map a sequence to a sequence are called *discrete-time systems*.

Some important classes of sequences and discrete-time systems have physical interpretations. For example, restrictions of sequences to the normed vector spaces  $\ell^2(\mathbb{Z})$  and  $\ell^\infty(\mathbb{Z})$  correspond to the physical phenomena of finite energy and boundedness. When the discretization of time is to evenly-spaced points, the constancy of physical laws corresponds to a shift-invariance property for discrete-time

systems. Linearity and shift invariance allow a system to be described uniquely by *convolution* with the system's *impulse response*. Once the convolution operation is defined, spectral theory allows us to construct an appropriate Fourier transform. Shift-invariant systems, convolution, and the discrete-time Fourier transform also have myriad uses that need not have a physical underpinning.

The above discussion implicitly assumed that the underlying domain, time, is infinite. In practice we glimpse a finite portion of time. Although it might seem that dealing with finite time would be easier, the tools we develop do not handle a beginning or an end to time. We discuss handling a finite amount of data throughout the chapter.

## 2.1 Introduction

While sequences in the real world are often one-sided infinite (they start at some initial point and then go on), for mathematical convenience, we look at them as two-sided infinite:<sup>33</sup>

$$x = [\dots \ x_{-2} \ x_{-1} \ \boxed{x_0} \ x_1 \ x_2 \ \dots]^T. \quad (2.1)$$

For example, if you measure some physical quantity every day at some fixed time (for example, the temperature in degrees at noon in front of your house), you obtain a sequence starting at time 0 (say January 14th) and continuing to infinity,

$$x = [\dots \ 0 \ \boxed{32} \ 29 \ 30 \ \dots]^T.$$

Implicit in the index is the fact that  $x_n$  corresponds to the temperature (at noon) on the  $n$ th day. A sequence is also known under the names discrete-time signal (in signal processing) or time series (in statistics); mathematically, these are all vectors, most often in an infinite-dimensional Hilbert space.

In real life, we observe only a finite portion of an infinite-length sequence. Moreover, computations are always done on finite inputs. For example, consistent temperature recordings started in the 18th century and necessarily stop at the present time, producing a sequence of length  $N$  for some finite  $N \in \mathbb{N}$ :

$$x = [\boxed{x_0} \ x_1 \ x_2 \ \dots \ x_{N-1}]^T. \quad (2.2)$$

Having only this data but methods that apply to all time, what do we do about days with no measurements? In effect, we are forced to assign some values. While we are limited only by our imaginations, two techniques stand out.

The first technique is to set  $x_n = 0$  for all  $n$  outside of  $\{0, 1, \dots, N-1\}$ . This is natural because, for any subsequent computation that uses  $x_n$  values linearly, this extension by zeros is equivalent to simply omitting measurements that are not available. However, the results are the same as if more data were available only when the signal is zero everywhere outside of  $\{0, 1, \dots, N-1\}$ .

<sup>33</sup>The boxing of the time origin is intended to serve as a reference point, essential when dealing with infinite vectors/matrices.

A second, less obvious technique is to extend the signal circularly:<sup>34</sup> periodize the finite-length sequence, treating the observed values as one period of a periodic sequence of period  $N \in \mathbb{N}$ :

$$x = [\dots \underbrace{x_0 \ x_1 \ \dots \ x_{N-1}}_{\text{one period}} \ x_0 \ x_1 \ \dots]^T. \quad (2.3)$$

While finite-length sequences in (2.2) and infinite-length periodic ones in (2.3) have a fundamentally different character, we will use the same types of tools to analyze them. Techniques designed explicitly for finite-length sequences are mathematically rooted in treating the sequence as one period of an infinite-length periodic sequence. The consequences of this implicit periodization are central in digital signal processing.

These considerations allow us to define two broad classes of sequences for which to develop our tools:

- (i) *Infinite-length sequences* are the vector space  $\mathbb{C}^{\mathbb{Z}}$  of sequences with domain  $\mathbb{Z}$ , as defined in (1.17b). The support of a sequence may be a proper subset of  $\mathbb{Z}$ ; for example, we will often consider infinite-length sequences that are nonzero only at nonnegative times.
- (ii) *Finite-length sequences*, without loss of generality, have support in  $\{0, 1, \dots, N-1\}$ . The tools we will develop do not treat the vector space of finite-length sequences as  $\mathbb{C}^N$  generically, but rather as sequences defined on a circular domain.

#### EXAMPLE 2.1 (SEQUENCES)

- (i) *Infinite-length sequences*: The geometric sequence (P1.65-2) with  $t = 1/2$ ,

$$x_n = \left(\frac{1}{2}\right)^n, \quad n \in \mathbb{Z}, \quad \text{or} \quad (2.4a)$$

$$x = [\dots \ 4 \ 2 \ \boxed{1} \ \frac{1}{2} \ \frac{1}{4} \ \dots]^T, \quad (2.4b)$$

is of infinite length and does not have finite  $\ell^1$ ,  $\ell^2$ , or  $\ell^\infty$  norm. If we made it nonzero only for  $n \geq 0$ , all these norms would be finite.

- (ii) *Finite-length sequences*: A sequence obtained by observing  $N$  tosses of a fair coin, recording a 0 for heads and 1 for tails,

$$x = [\boxed{0} \ 0 \ 1 \ 1 \ 0 \ 1 \ 0 \ \dots \ 0]^T,$$

is of finite length. There is no extension of this sequence outside of the  $N$  observed values that is particularly natural.

<sup>34</sup>Another name for a circular extension is a periodic extension.

A sinusoidal function sampled at  $N$  samples per period,

$$x_n = \sin\left(\frac{2\pi}{N}n + \theta\right), \quad n \in \mathbb{Z}, \quad \text{or,}$$

$$x = [\dots \underbrace{\sin(\theta) \sin\left(\frac{2\pi}{N} + \theta\right) \dots \sin\left(\frac{2\pi}{N}(N-1) + \theta\right)}_{\text{one period}} \sin(\theta) \dots]^T,$$

is an infinite-length periodic sequence. Taking  $N$  samples

$$x = [\boxed{x_0} \quad x_1 \quad x_2 \quad \dots \quad x_{N-1}]^T$$

gives a finite-length sequence for which circular extension is quite natural.

Given sequences (signals, vectors), one can apply operators (systems, filters). These map input sequences into output ones, and since they involve discrete-time sequences, they are usually called *discrete-time systems* (operators). Among them, linear ones are the most common. Even more restricted is the class of *linear shift-invariant* systems (filters, defined later in the chapter), an example of which is the moving average filter:

**EXAMPLE 2.2 (MOVING AVERAGE FILTER)** Consider our temperature example, and assume we want to detect seasonal trends. The day-by-day variation might be too erratic, and thus, we compute a local average

$$y_n = \frac{1}{N} \sum_{k=-(N-1)/2}^{(N-1)/2} x_{n-k}, \quad n \in \mathbb{Z}, \quad (2.5)$$

where  $N$  is a small, odd integer. The local average reduces daily variations; this simple filter is linear and shift invariant (defined later in the chapter), since at all  $n$ , the same local averaging is performed.

## Chapter Outline

The next several sections follow the progression of topics in this brief introduction: In Section 2.2, we start by formally defining the various types of sequences we discussed above. Section 2.3 considers linear discrete-time systems, especially of the shift-invariant kind, which correspond to difference equations, the discrete-time analogue of differential equations. Next, in Sections 2.4–2.6, we develop the tools to analyze discrete-time sequences and systems, in particular the discrete-time Fourier transform, the  $z$ -transform, and the discrete Fourier transform. We discuss the fundamental result relating filtering to multiplication in Fourier domain—the convolution property. Section 2.7 looks into discrete-time sequences and systems that operate with different rates—multirate systems, which are key for filter-bank development in later chapters. This is followed by discrete stochastic processes and systems in Section 2.8, while important algorithms for discrete-time processing,

such as the fast Fourier transform, are covered in Section 2.9. Appendix 2.A.1 lists basic elements of complex analysis, while Appendix 2.B discusses some elements of algebra, in particular, polynomial sequences.

*Notation used in this chapter:* We assume sequences to be complex in general, at the risk of a bit more cumbersome notation at times. Thus, Hermitian transposition is used often. We will be using  $\|\cdot\|$  to denote the 2 norm; any other norm, such as the 1 norm,  $\|\cdot\|_1$ , will be explicitly specified.  $\square$

## 2.2 Sequences

### 2.2.1 Infinite-Length Sequences

The set of sequences in (2.1), where  $x_n$  is either real or complex, together with vector addition and scalar multiplication, forms a vector space (see Definition 1.1). The inner product between two infinite-length sequences is defined in (1.20b) and induces the standard  $\ell^2$  (or Euclidean) norm (1.23b). Other norms of interest are the  $\ell^1$  norm from (1.36a) with  $p = 1$ , and the  $\infty$  norm from (1.36b).

As opposed to generic infinite-dimensional spaces, where ordering of indices does not matter in general, discrete-time sequences belong to an infinite-dimensional space where ordering of indices is important since it represents time. Note that in some instances later in the book, we will be dealing with vectors of sequences, for example,  $x = [x_0 \ x_1]^T$  where  $x_0$  and  $x_1$  are sequences as well. We now look into a few spaces of interest.

#### Sequence Spaces

**Space of Square-Summable Sequences  $\ell^2(\mathbb{Z})$**  The constraint of a finite square norm is necessary for turning the vector space  $\mathbb{C}^{\mathbb{Z}}$  defined in (1.17b) into the Hilbert space of *finite-energy* sequences  $\ell^2(\mathbb{Z})$ . This space affords a geometric view; we now recall a few such geometric facts from Chapter 1:

- (i) The inner product between two vectors is

$$\langle x, y \rangle = \|x\| \|y\| \cos \theta,$$

where  $\theta$  is the angle between the two infinite-length sequences (vectors).

- (ii) As in Definition 1.8, if the inner product is zero,

$$\langle x, y \rangle = 0,$$

the sequences are said to be orthogonal to each other.

- (iii) As in (1.61), given a unit-norm sequence  $y$ ,  $\|y\| = 1$ ,

$$\hat{x} = \langle x, y \rangle y$$

is the orthogonal projection of the sequence  $x$  onto the space spanned by the sequence  $y$ .

**Space of Bounded Sequences**  $\ell^\infty(\mathbb{Z})$  A looser constraint than finite energy is to bound the magnitude of the samples. The space of bounded sequences contains all sequences  $x_n$  such that, for some finite  $M$ ,  $|x_n| \leq M$  for all  $n \in \mathbb{Z}$ . This space is denoted  $\ell^\infty(\mathbb{Z})$  since it consists of sequences with finite  $\ell^\infty$  norm.

**Space of Absolutely-Summable Sequences**  $\ell^1(\mathbb{Z})$  A more restrictive constraint than finite energy is to require absolute summability (remember that  $\ell^1(\mathbb{Z}) \subset \ell^2(\mathbb{Z})$  from (1.37)). By definition, sequences in  $\ell^1(\mathbb{Z})$  have a finite  $\ell^1$  norm.

EXAMPLE 2.3 (SEQUENCE SPACES) Revisiting the geometric sequence such as the one in (2.4), we see that for  $\alpha \in \mathbb{R}$ ,

$$x_n = \begin{cases} 0, & \text{for } n < 0; \\ \alpha^n, & \text{for } n \geq 0 \end{cases} \quad (2.6)$$

is in the following spaces:

$$\text{For } \left\{ \begin{array}{l} |\alpha| < 1 \\ |\alpha| = 1 \\ |\alpha| > 1 \end{array} \right\}, \quad x \in \left\{ \begin{array}{l} \ell^2(\mathbb{Z}), \ell^1(\mathbb{Z}), \ell^\infty(\mathbb{Z}) \\ \ell^\infty(\mathbb{Z}) \\ \text{none} \end{array} \right\}.$$

### Special Sequences

We now introduce the sequences most often used in the book.

**Kronecker Delta Sequence** The simplest nonzero sequence is the *Kronecker delta* sequence,

$$\delta_n = \begin{cases} 1, & \text{for } n = 0; \\ 0, & \text{otherwise,} \end{cases} \quad n \in \mathbb{Z}, \quad \text{or,} \quad (2.7a)$$

$$\delta = [\dots \ 0 \ \boxed{1} \ 0 \ \dots]^T. \quad (2.7b)$$

Shifting the single 1 in the sequence to position  $k$  gives what is called the Kronecker delta sequence at location  $k$ , which is  $\delta_{n-k}$ . The set of Kronecker delta sequences  $\{\delta_{n-k}\}_{k \in \mathbb{Z}}$  along the discrete time line forms an orthonormal basis for  $\ell^2(\mathbb{Z})$ ; we called it the standard basis in Chapter 1. Table 2.1 lists some properties of the Kronecker delta sequence. (The shifting property uses convolution, which is defined in (2.59).)

**Sinc Function and Sequences** The *sinc function* appears frequently in signal processing and approximation. It is defined as

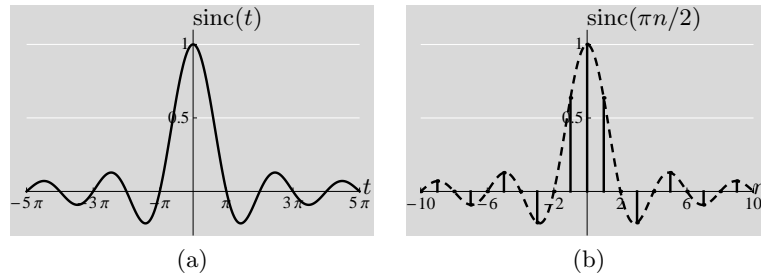
$$\text{sinc}(t) = \begin{cases} (\sin t)/t, & \text{for } t \neq 0; \\ 1, & \text{for } t = 0. \end{cases} \quad (2.8a)$$

Evaluation of  $\lim_{t \rightarrow 0} \text{sinc}(t)$  using l'Hôpital's rule confirms the continuity of the function. Scaling the sinc function with  $1/\sqrt{\pi}$  makes it of unit norm, that is,

$$\left\| \frac{1}{\sqrt{\pi}} \text{sinc}(t) \right\| = 1. \quad (2.8b)$$



Kronecker delta sequence	
Normalization	$\sum_{n \in \mathbb{Z}} \delta_n = 1$
Sifting	$\sum_{n \in \mathbb{Z}} x_{n_0-n} \delta_n = \sum_{n \in \mathbb{Z}} \delta_{n_0-n} x_n = x_{n_0}$
Shifting	$x_n * \delta_{n-n_0} = x_{n-n_0}$
Sampling	$x_n \delta_n = x_0 \delta_n$
Restriction	$x_n \delta_n = 1_{\{0\}} x$

**Table 2.1:** Properties of the Kronecker delta sequence.**Figure 2.1:** (a) The sinc function  $\text{sinc}(t)$ . (b) The sinc sequence  $\text{sinc}(\pi n/2)$ .

The sinc function is zero at  $t_n = n\pi$ , for  $n \neq 0$ ; together with the value at  $t = 0$ , this gives

$$\text{sinc}(n\pi) = \delta_n, \quad n \in \mathbb{Z}. \quad (2.8c)$$

The sinc function is illustrated in Figure 2.1(a).

For any positive  $T$ , we can obtain a sinc sequence

$$\frac{1}{\sqrt{T}} \text{sinc}(\pi n/T) = \frac{1}{\sqrt{T}} \frac{\sin(\pi n/T)}{\pi n/T}. \quad (2.9)$$

This sequence is of unit norm and is in  $\ell^\infty(\mathbb{Z})$  and in  $\ell^2(\mathbb{Z})$ . It is not in  $\ell^1(\mathbb{Z})$  since it decays as  $1/n$  (see Example 1.8, illustrating the inclusion property (1.37) of  $\ell^p$  spaces). It is zero at nonzero integers  $n/T$ ; this is illustrated in Figure 2.1(b) for  $T = 2$ .

**Heaviside Sequence** The *Heaviside* or *unit-step* sequence is defined as

$$u_n = \begin{cases} 1, & \text{for } n \in \mathbb{N}; \\ 0, & \text{otherwise,} \end{cases} \quad n \in \mathbb{Z}, \quad \text{or,} \quad (2.10a)$$

$$u = [\dots 0 \boxed{1} 1 \dots]^T. \quad (2.10b)$$

This sequence is bounded by 1, so it belongs to  $\ell^\infty(\mathbb{Z})$ . It belongs to neither  $\ell^1(\mathbb{Z})$  nor  $\ell^2(\mathbb{Z})$ . The Kronecker delta and Heaviside sequences are related via

$$u_n = \sum_{k=-\infty}^n \delta_k.$$

Pointwise multiplication by the Heaviside sequence implements the domain restriction operator (1.57) for restriction from all the integers to just the nonnegative integers:

$$1_{\mathbb{N}} x = \begin{cases} x_n, & \text{for } n \in \mathbb{N}; \\ 0, & \text{otherwise} \end{cases} = u_n x_n, \quad n \in \mathbb{Z}.$$

From this we can also build other domain restriction operators. For example, domain restriction to  $\{n_0, n_0 + 1, \dots, n_1\}$  is achieved with a difference of two shifted Heaviside sequences:

$$1_{\{n_0, \dots, n_1\}} x = (u_{n-n_0} - u_{n-n_1-1}) x_n = \begin{cases} x_n, & \text{for } n \in \{n_0, \dots, n_1\}; \\ 0, & \text{otherwise.} \end{cases} \quad (2.11)$$

**Box and Window Sequences** For any positive integer  $n_0$ , the (unnormalized) *right-sided box sequence* is defined as

$$w_n = \begin{cases} 1, & \text{for } 0 \leq n \leq n_0 - 1; \\ 0, & \text{otherwise,} \end{cases} \quad n \in \mathbb{Z}, \quad \text{or,} \quad (2.12a)$$

$$w = [\dots 0 \underbrace{1 \ 1 \ \dots \ 1}_{n_0} 0 \ \dots]^T. \quad (2.12b)$$

For odd  $n_0$ , the *centered* and *normalized* box sequence is defined as

$$w_n = \begin{cases} 1/\sqrt{n_0}, & \text{for } |n| \leq (n_0 - 1)/2; \\ 0, & \text{otherwise,} \end{cases} \quad n \in \mathbb{Z}, \quad \text{or,} \quad (2.13a)$$

$$w = \left[ \dots 0 \quad \frac{1}{\sqrt{n_0}} \quad \dots \quad \boxed{\frac{1}{\sqrt{n_0}}} \quad \dots \quad \frac{1}{\sqrt{n_0}} \quad 0 \quad \dots \right]^T. \quad (2.13b)$$

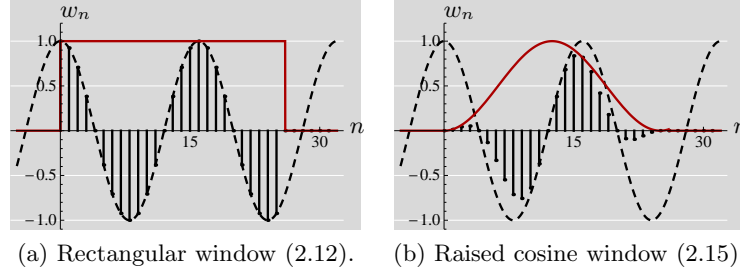
Box sequences are also called *rectangular window* sequences.

Often, a finite-length sequence is treated as a glimpse of an infinite-length sequence. One way to state this is by using pointwise multiplication with a window sequence. Multiplying an arbitrary sequence  $x$  with the right-sided window  $w$  given in (2.12), we obtain a windowed version of  $x$ :

$$\hat{x}_n = x_n w_n, \quad n \in \mathbb{Z}, \quad \text{or,} \quad (2.14a)$$

$$\hat{x} = [\dots 0 \quad \boxed{x_0} \quad x_1 \quad \dots \quad x_{n_0-1} \quad 0 \quad \dots]^T. \quad (2.14b)$$

With this use of an unnormalized rectangular window,  $\hat{x}_n$  equals  $x_n$  for  $n \in \{0, 1, \dots, n_0 - 1\}$  and is zero otherwise. We sometimes study  $x$  through the finite-length sequence  $\hat{x}$  that coincides with  $x$  over a window of interest.



**Figure 2.2:** A sinusoidal sequence  $x_n = \sin((\pi/8)n + \pi/2)$  (dashed plots) and its windowed versions  $w_n x_n$  (stem plots) with two different windows of length  $n_0 = 26$  (solid plots).

How good is the window we just used? For example, if  $x$  is smooth, its windowed version  $\hat{x}$  is not because of the abrupt boundaries of the rectangular window. We might thus decide to use a different window to smooth the boundaries, an example of which we now discuss. We look at other commonly used windows in Chapter 8, Exercise 8.7.

**EXAMPLE 2.4 (WINDOWS)** Consider an infinite-length sinusoidal sequence of frequency  $\omega_0$  and phase  $\theta$ ,

$$x_n = \sin(\omega_0 n + \theta),$$

and the following two windows:

- (i) a rectangular, length- $n_0$  window, as in (2.12); and
- (ii) a *raised cosine* window,<sup>35</sup> also of length  $n_0$ :

$$w_n = \begin{cases} \frac{1}{2}(1 - \cos \frac{2\pi n}{n_0 - 1}), & \text{for } 0 \leq n \leq n_0 - 1; \\ 0, & \text{otherwise.} \end{cases} \quad (2.15)$$

The raised cosine window tapers off smoothly at the boundaries, while the rectangular one does not. The trade-off between the two windows is obvious from Figure 2.2: the rectangular window does not modify the sequence inside the window, but has abrupt transitions at the boundary, while the raised cosine window has smooth transitions at the boundary, but at the price of modifying the sequence inside the window.

### Deterministic Correlation

We now discuss two operations on sequences, both deterministic, that appear throughout the chapter. Stochastic versions of both operations will be given in Section 2.8.1.

<sup>35</sup>This is also known as a Hann or Hanning window, after Julius von Hann.

**Deterministic Autocorrelation** The *deterministic autocorrelation*  $a$  of a sequence  $x$  is

$$a_n = \sum_{k \in \mathbb{Z}} x_k x_{k-n}^* = \langle x_k, x_{k-n} \rangle_k, \quad (2.16)$$

where the final expression introduces a notation in which the variable over which to sum,  $k$ , is explicitly included in the inner product notation. This simplifies our discussion because we can use  $x_{k-n}$  instead of a new symbol for this time-reversed and shifted version of  $x$ . The deterministic autocorrelation satisfies

$$a_n = a_{-n}^*, \quad (2.17a)$$

$$a_0 = \sum_{k \in \mathbb{Z}} |x_k|^2 = \|x\|^2, \quad (2.17b)$$

the proof of which is left for Exercise 2.5. The deterministic autocorrelation measures the similarity of a sequence with respect to shifts of itself, and it is Hermitian symmetric as in (2.17a). For a real  $x$ ,

$$a_n = \sum_{k \in \mathbb{Z}} x_k x_{k-n} = a_{-n}. \quad (2.17c)$$

When we need to specify the sequence involved, we write  $a_{x,n}$ .

**EXAMPLE 2.5 (DETERMINISTIC AUTOCORRELATION OF A FINITE-LENGTH SEQUENCE)**  
Assume  $x$  is the box sequence from (2.13a) with  $n_0 = 3$ , that is, a constant sequence of length 3 and height  $1/\sqrt{3}$ . Using (2.16), we compute its deterministic autocorrelation to be:

$$a_x = \left[ \dots \quad 0 \quad \frac{1}{3} \quad \frac{2}{3} \quad \boxed{1} \quad \frac{2}{3} \quad \frac{1}{3} \quad 0 \quad \dots \right]^T. \quad (2.18)$$

This sequence is clearly symmetric, satisfying (2.17c).

**Deterministic Crosscorrelation** The *deterministic crosscorrelation*  $c$  of two sequences  $x$  and  $y$  is

$$c_n = \sum_{k \in \mathbb{Z}} x_k y_{k-n}^* = \langle x_k, y_{k-n} \rangle_k, \quad (2.19)$$

and is written as  $c_{x,y,n}$  to specify the sequences involved. It satisfies

$$c_{x,y,n} = \left( \sum_{k \in \mathbb{Z}} y_{k-n} x_k^* \right)^* \stackrel{(a)}{=} \left( \sum_{m \in \mathbb{Z}} y_m x_{m+n}^* \right)^* = c_{y,x,-n}^*, \quad (2.20a)$$

where (a) follows from change of variable  $m = k - n$  (see also Exercise 2.5). For real  $x$  and  $y$ ,

$$c_{x,y,n} = \sum_{k \in \mathbb{Z}} x_k y_{k-n} = c_{y,x,-n}. \quad (2.20b)$$

**EXAMPLE 2.6 (DETERMINISTIC CROSSCORRELATION OF TWO FINITE-LENGTH SEQUENCES)**

Assume  $x$  is the box sequence from (2.13a) with  $n_0 = 3$ , as in Example 2.5, and

$$y = \left[ \dots \quad 0 \quad 0 \quad \boxed{\sqrt{2/3}} \quad 1/\sqrt{3} \quad 0 \quad 0 \quad \dots \right]^T.$$

Using (2.19), we compute the deterministic crosscorrelations

$$c_{x,y} = \left[ \dots \quad 0 \quad \frac{1}{3} \quad \frac{1+\sqrt{2}}{3} \quad \boxed{\frac{1+\sqrt{2}}{3}} \quad \frac{\sqrt{2}}{3} \quad 0 \quad 0 \quad \dots \right]^T, \quad (2.21a)$$

$$c_{y,x} = \left[ \dots \quad 0 \quad 0 \quad \frac{\sqrt{2}}{3} \quad \boxed{\frac{1+\sqrt{2}}{3}} \quad \frac{1+\sqrt{2}}{3} \quad \frac{1}{3} \quad 0 \quad \dots \right]^T; \quad (2.21b)$$

thus, clearly, (2.20b) is satisfied.

**Deterministic Autocorrelation of Vector Sequences** Consider a vector of  $N$  sequences, that is, an infinite matrix whose  $(k+1)$ st row is the sequence  $x_k = [\dots \quad x_{k,-1} \quad \boxed{x_{k,0}} \quad x_{k,1} \quad \dots]$ ,

$$x = [x_0 \quad x_1 \quad \dots \quad x_{N-1}]^T.$$

Its deterministic autocorrelation is a sequence of matrices given by

$$A_n = \begin{bmatrix} a_{0,n} & c_{0,1,n} & \dots & c_{0,N-1,n} \\ c_{1,0,n} & a_{1,n} & \dots & c_{1,N-1,n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{N-1,0,n} & c_{N-1,1,n} & \dots & a_{N-1,n} \end{bmatrix}, \quad (2.22)$$

that is, a matrix with individual sequence deterministic autocorrelations  $a_{i,n}$  on the diagonal and the pairwise deterministic crosscorrelations  $c_{i,k,n}$  off the diagonal, for  $i, k = 0, 1, \dots, N-1$ ,  $i \neq k$ . Because of (2.17a) and (2.20a),  $A_n$  satisfies

$$A_n = \begin{bmatrix} a_{0,n} & c_{0,1,n} & \dots & c_{0,N-1,n} \\ c_{0,1,-n}^* & a_{1,n} & \dots & c_{1,N-1,n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{0,N-1,-n}^* & c_{1,N-1,-n}^* & \dots & a_{N-1,n} \end{bmatrix} = A_{-n}^*, \quad (2.23a)$$

that is, it is a Hermitian matrix (see (1.221a)). For a real  $x$ , it is a symmetric matrix,

$$A_n = A_{-n}^T. \quad (2.23b)$$

**EXAMPLE 2.7 (DETERMINISTIC AUTOCORRELATION OF A VECTOR SEQUENCE)**

Assume we are given a vector of two sequences  $x = [x_0 \quad x_1]^T$  with  $x_0 = x$  and  $x_1 = y$  from Example 2.6. Its deterministic autocorrelation is then

$$A_n = \begin{bmatrix} a_{0,n} & c_{0,1,n} \\ c_{1,0,n} & a_{1,n} \end{bmatrix}.$$

We have already computed three out of four entries in the above matrix: the deterministic autocorrelation sequence  $a_0 = a_x$  from (2.18) and the deterministic crosscorrelation sequences  $c_{0,1} = c_{x,y}$  from (2.21a) and  $c_{1,0} = c_{y,x}$  from (2.21b). The only entry left to compute is the deterministic autocorrelation sequence  $a_y$ :

$$a_y = \left[ \dots \quad 0 \quad \frac{\sqrt{2}}{3} \quad \boxed{1} \quad \frac{\sqrt{2}}{3} \quad 0 \quad \dots \right]^T, \quad (2.24)$$

again, a symmetric sequence. Because of what we already computed in Examples 2.5 and 2.6, that is, the deterministic autocorrelation  $a_x$  is symmetric, and  $c_{x,y,n} = c_{y,x,-n}$ , (2.23b) is satisfied. A few of these  $A_n$  are:

$$\left\{ \dots, \begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} \frac{2}{3} & \frac{1+\sqrt{2}}{3} \\ \frac{\sqrt{2}}{3} & \frac{\sqrt{2}}{3} \end{bmatrix}, \boxed{\begin{bmatrix} 1 & \frac{1+\sqrt{2}}{3} \\ \frac{1+\sqrt{2}}{3} & 1 \end{bmatrix}}, \begin{bmatrix} \frac{2}{3} & \frac{\sqrt{2}}{3} \\ \frac{\sqrt{2}}{3} & \frac{\sqrt{2}}{3} \end{bmatrix}, \begin{bmatrix} \frac{1}{3} & 0 \\ \frac{1}{3} & 0 \end{bmatrix}, \dots \right\}.$$

### 2.2.2 Finite-Length Sequences

Finite-length sequences as in (2.2) are those with the domain

$$n \in \{0, 1, \dots, N-1\}$$

for some positive integer  $N$ . A finite-length sequence can be seen either as an infinite-length sequence that happens to take nonzero values only inside  $\{0, 1, \dots, N-1\}$  or as a period of a periodic sequence with

$$x_{n+kN} = x_n, \quad k \in \mathbb{Z}. \quad (2.25)$$

#### Sequence Spaces

In the case of a periodic sequence, it is useful to think of the domain itself as wrapping around into a circle, with  $N-1$  next to 0. On this discrete circle domain, incrementing the time index is not ordinary addition but rather addition modulo  $N$ , so we could refer to the domain as  $\mathbb{Z}_N$  and to the vector space of these sequences as  $\mathbb{C}^{\mathbb{Z}_N}$ . We do not actually adopt the notation  $\mathbb{C}^{\mathbb{Z}_N}$  because the standard vector space operations (see Definition 1.1) are the same as for  $\mathbb{C}^N$ .

Differences between periodic sequences (those defined on a circular domain) and infinite sequences with finite support emerge with operations that we introduce later. For periodic sequences, there is a circular form of convolution. Applying spectral theory to this convolution leads to the discrete Fourier transform. As part of a more general theory, other convolution operators would lead to different Fourier transforms; more details on this topic can be found in *Further Reading*.

#### Special Sequences

**Periodic Kronecker Delta Sequences** A periodic version of the Kronecker delta sequence is obtained by adding all shifts of  $\delta$  by integer multiples of  $N$ :

$$\varphi_n = \sum_{\ell \in \mathbb{Z}} \delta_{n-\ell N}, \quad n \in \mathbb{Z}.$$

The resulting sequence is

$$\begin{aligned}\varphi_n &= \begin{cases} 1, & \text{for } n = \ell N, \ell \in \mathbb{Z}; \\ 0, & \text{otherwise,} \end{cases} \quad n \in \mathbb{Z}, \quad \text{or,} \\ \varphi &= [\dots 0 \underbrace{1 \ 0 \ \dots \ 0}_N \ 1 \ 0 \ \dots]^T.\end{aligned}$$

The set of  $N$  sequences generated from this  $\varphi$  by shifts in  $\{0, 1, \dots, N-1\}$  span the space of  $N$ -periodic sequences.

**Complex Exponential Sequences** As we will see in Section 2.6, the complex exponential sequences form a natural basis for  $N$ -periodic sequences. These are the  $N$  sequences  $\varphi_k$ ,  $k \in \{0, 1, \dots, N-1\}$ , given by

$$\varphi_{k,n} = \frac{1}{\sqrt{N}} e^{j(2\pi/N)kn}, \quad k \in \{0, 1, \dots, N-1\}, \quad n \in \mathbb{Z}. \quad (2.26)$$

Each of these sequences is periodic with period  $N$ . In Solved Exercise 2.1, we explore a few properties of complex exponential sequences.

### 2.2.3 Multidimensional Sequences

#### Two-Dimensional Sequences

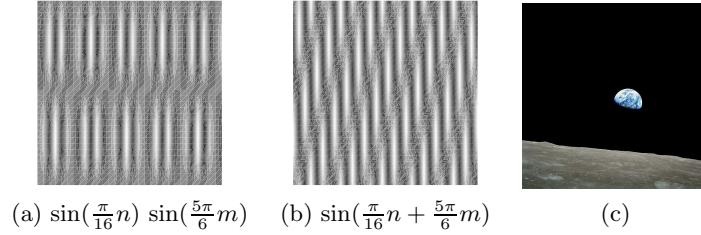
Today, one of the most widespread devices is the digital camera. In our notation, a digital picture is a two-dimensional sequence,  $x_{n,m}$ . It can be seen either as an infinite-length sequence with a finite number of nonzero samples,

$$x_{n,m}, \quad n, m \in \mathbb{Z}, \quad (2.27)$$

or as a sequence with domain  $n \in \{0, 1, \dots, N-1\}$ ,  $m \in \{0, 1, \dots, M-1\}$ , conveniently expressed as a matrix:

$$x = \begin{bmatrix} x_{0,0} & x_{0,1} & \dots & x_{0,M-1} \\ x_{1,0} & x_{1,1} & \dots & x_{1,M-1} \\ \vdots & \vdots & & \vdots \\ x_{N-1,0} & x_{N-1,1} & \dots & x_{N-1,M-1} \end{bmatrix}. \quad (2.28)$$

While circularly extending the image at the borders is perhaps not natural (the top of the image appears next to the bottom), it is the extension that leads to the use of the discrete Fourier transform, as we will see later in the chapter. Each element  $x_{n,m}$  is called a *pixel*, and the total image has  $NM$  pixels. In reality, for  $x_{n,m}$  to represent a color image, it must have more than one component; often, red, green and blue components are used (RGB space). Figure 2.3 gives examples of higher-dimensional sequences.



**Figure 2.3:** Multidimensional sequences. (a) Two-dimensional separable sinusoidal sequence. (b) Two-dimensional nonseparable sinusoidal sequence. (c) Earth visible above the lunar surface, taken by Apollo 8 crew member Bill Anders on December 24, 1968. This could be considered a two-dimensional sequence if the image were gray scale representing the intensity, or a higher-dimensional sequence depending on how color is represented.

Sequence Space	Symbol	Finite Norm
Absolutely-summable	$\ell^1(\mathbb{Z}^2)$	$\ x\ _1 = \sum_{n,m \in \mathbb{Z}}  x_{n,m} $
Square-summable/finite-energy	$\ell^2(\mathbb{Z}^2)$	$\ x\  = \left( \sum_{n,m \in \mathbb{Z}}  x_{n,m} ^2 \right)^{1/2}$
Bounded	$\ell^\infty(\mathbb{Z}^2)$	$\ x\ _\infty = \sup_{n,m \in \mathbb{Z}}  x_{n,m} $

**Table 2.2:** Norms and two-dimensional sequence spaces.

**Sequence Spaces** The spaces we introduced in one dimension generalize to multiple dimensions; for example, in two dimensions the inner product of two sequences  $x$  and  $y$  is

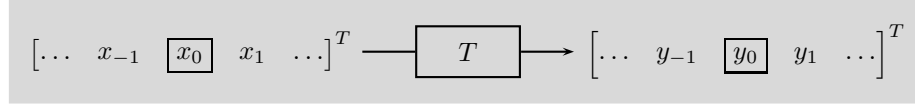
$$\langle x, y \rangle = \sum_{n \in \mathbb{Z}} \sum_{m \in \mathbb{Z}} x_{n,m} y_{n,m}^*, \quad (2.29)$$

while the  $\ell^2$  norm and the appropriate space  $\ell^2(\mathbb{Z}^2)$  are given in Table 2.2, together with other relevant norms and spaces. For example, a digital picture, having finite size and pixel values that are bounded, clearly belongs to all three spaces defined in Table 2.2. Infinite-length multidimensional sequences, on the other hand, can be harder to analyze.

**EXAMPLE 2.8 (NORMS OF TWO-DIMENSIONAL SEQUENCES)** Consider the sequence

$$x_{n,m} = \frac{1}{2^n 3^m}, \quad n, m \in \mathbb{N}.$$



**Figure 2.4:** A discrete-time system.

Its  $\ell^2$  norm can be evaluated as<sup>36</sup>

$$\langle x, x \rangle = \sum_{n \in \mathbb{N}} \sum_{m \in \mathbb{N}} \frac{1}{4^n} \frac{1}{9^m} = \left( \sum_{n \in \mathbb{N}} \frac{1}{4^n} \right) \left( \sum_{m \in \mathbb{N}} \frac{1}{9^m} \right) = \frac{4}{3} \cdot \frac{9}{8} = \frac{3}{2},$$

yielding  $\|x\| = \sqrt{3/2}$ . Similarly,  $\|x\|_1 = 3$  and  $\|x\|_\infty = 1$ .

## 2.3 Systems

Discrete-time systems are operators having discrete-time sequences as their inputs and outputs. Among all discrete-time systems, we will concentrate on those that are linear and shift-invariant. This subclass is both important in practice and amenable to easy analysis. The moving average filter in (2.5) is such a linear, shift-invariant system. After an introduction to difference equations, which are natural descriptions of discrete-time systems, we study linear, shift-invariant systems in detail.

### 2.3.1 Discrete-Time Systems and Their Properties

A discrete-time system is an operator  $T$  that maps an input sequence  $x \in V$  into an output sequence  $y \in V$ ,

$$y = T(x), \quad (2.30)$$

as shown in Figure 2.4. As we have seen in the previous section, the sequence space  $V$  is typically  $\ell^2(\mathbb{Z})$  or  $\ell^\infty(\mathbb{Z})$ . At times, the input or the output is in a subspace of such spaces.

#### Types of Systems

Discrete-time systems can have a number of useful properties. We will encounter these same properties in Chapter 3 as well. After defining key properties, we will illustrate them on certain basic systems.

n

<sup>36</sup>We interchange summations freely, which can be done because each one-dimensional sequence involved is absolutely summable. When this is not the case, one has to be careful, as discussed in Appendix 1.A.2.

**Linear Systems** Similarly to Definition 1.17, linearity<sup>37</sup> combines two properties: additivity (the output of a sum of sequences is the sum of the outputs of the sequences) and scaling (the output of a scaled sequence is the scaled output of the sequence).

**DEFINITION 2.1 (LINEAR SYSTEM)** A discrete-time system  $T$  is called linear when, for any inputs  $x$  and  $y$  and any  $\alpha, \beta \in \mathbb{C}$ ,

$$T(\alpha x + \beta y) = \alpha T(x) + \beta T(y). \quad (2.31)$$

The function  $T$  is thus a linear operator, and we write (2.30) as

$$y = T x. \quad (2.32)$$

We will often use a matrix representation for a linear system, especially when the structure of the matrix reveals properties of the system.

As discussed in Section 1.5.5, a linear operator has a unique matrix representation once bases have been chosen for the domain and codomain of the operator. Throughout this chapter, matrix representations of linear systems will be with respect to the standard basis (the Kronecker delta sequence and its shifts) for both the inputs and outputs. The general form of the matrix representation then follows from (1.152): column  $k$  holds the output that results from taking the shifted Kronecker delta sequence  $\delta_{n-k}$  as the input. To be more explicit, for each  $k \in \mathbb{Z}$ , let input  $x^{(k)}$  result in output  $y^{(k)}$ , where

$$x_n^{(k)} = \delta_{n-k}, \quad n \in \mathbb{Z}.$$

Then the matrix representation of the system is

$$\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & y_{-1}^{(-1)} & y_{-1}^{(0)} & y_{-1}^{(1)} & \cdots \\ \cdots & y_0^{(-1)} & y_0^{(0)} & y_0^{(1)} & \cdots \\ \cdots & y_1^{(-1)} & y_1^{(0)} & y_1^{(1)} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}. \quad (2.33)$$

**Memoryless Systems** Certain simple systems are *instantaneous* in that they act only based on the current input sample. It follows that if two inputs agree at a time index  $k$ , the corresponding outputs must also agree at time index  $k$ . For a mathematical representation of memorylessness, we use the domain restriction operator defined in (1.57).

<sup>37</sup>In the engineering literature, linearity and *superposition principle* are often used interchangeably.

**DEFINITION 2.2 (MEMORYLESS SYSTEM)** A discrete-time system  $T$  is called memoryless when, for any integer  $k$  and inputs  $x$  and  $x'$ ,

$$1_{\{k\}} x = 1_{\{k\}} x' \Rightarrow 1_{\{k\}} T(x) = 1_{\{k\}} T(x'). \quad (2.34)$$

In a matrix representation of a linear and memoryless system, the matrix will be diagonal; we will illustrate this and other properties of matrix representations of linear systems in several examples shortly.

**Causal Systems** The output of a causal system at time index  $k$  depends on the input only up to time index  $k$ . It follows that if two inputs agree up to time  $k$ , the corresponding outputs must agree up to time  $k$ .

**DEFINITION 2.3 (CAUSAL SYSTEM)** A discrete-time system  $T$  is called causal when, for any integer  $k$  and inputs  $x$  and  $x'$ ,

$$1_{\{-\infty, \dots, k\}} x = 1_{\{-\infty, \dots, k\}} x' \Rightarrow 1_{\{-\infty, \dots, k\}} T(x) = 1_{\{-\infty, \dots, k\}} T(x'). \quad (2.35)$$

In a matrix representation of a linear and causal system, the matrix will be lower triangular.

Since a computation cannot depend on inputs that will only be provided in the future, causality can seem to be a property that is required of any implemented system. However, this view takes the concept of the time index representing time too literally. First, the discrete time index may represent something else entirely, like a physical location along a line; the data can then be processed in any order. Second, when the time index indeed represents time, the time origins of the input and output need not coincide; then, causality sometimes represents nothing more than a convenient convention for aligning time indexes of the input and output.

**Shift-Invariant Systems** In a shift-invariant system, shifting the input has the effect of shifting the output by the same amount:

**DEFINITION 2.4 (SHIFT-INVARIANT SYSTEM)** A discrete-time system  $T$  is called shift invariant when, for any integer  $k$  and input  $x$ ,

$$y = T(x) \Rightarrow y' = T(x'), \quad \text{where } x'_n = x_{n-k} \text{ and } y'_n = y_{n-k}. \quad (2.36)$$

In a matrix representation of a linear and shift-invariant system, the matrix will be Toeplitz.

Shift invariance (or, when it corresponds to time, time invariance) is often a desirable property. For example, an MP3 player should produce the same music

from the same file on Tuesday as on Monday. Moreover, *linear shift-invariant* (LSI) or linear time-invariant (LTI) systems have desirable mathematical properties. Much of the remainder of this section and Sections 2.4 and 2.5 are devoted to the powerful analysis techniques that apply to LSI systems. Sections 2.6 and 2.7 include variations on shift invariance and the corresponding techniques.

**Stable Systems** A critical property for a discrete-time system is its stability. While various definitions exist, they all require that the system remain well behaved when presented with a certain class of inputs. We define *bounded-input bounded-output* (BIBO) stability here, because it is both practical and easy to check in cases of interest.

**DEFINITION 2.5 (BIBO STABILITY)** A discrete-time system  $T$  is called bounded-input bounded-output stable when a bounded input  $x$  produces a bounded output  $y = T(x)$ :

$$x \in \ell^\infty(\mathbb{Z}) \quad \Rightarrow \quad y \in \ell^\infty(\mathbb{Z}). \quad (2.37)$$

In a matrix representation of a linear and BIBO-stable system, every row of the matrix will be absolutely summable. The corresponding result for LSI systems is developed fully in Section 2.3.3

The definition of BIBO stability involves the  $\ell^\infty$  norm, so we can see immediately that a system that is linear and BIBO stable is a bounded linear operator from  $\ell^\infty(\mathbb{Z})$  to  $\ell^\infty(\mathbb{Z})$ . The absolute-summability condition on the system that ensures BIBO stability also ensures that the system is a bounded linear operator from  $\ell^2(\mathbb{Z})$  to  $\ell^2(\mathbb{Z})$  (see Exercise TBD). Thus, when we limit attention to BIBO stable systems, we are able to use the various results for bounded linear operators on a Hilbert space that were developed in Chapter 1.

### Basic Systems

We now discuss a few basic discrete-time systems. These include some basic building blocks that we will use frequently. Their properties are summarized in Table 2.3.

**Shift** The shift-by-1, or *delay*, operator is defined as:

$$y_n = x_{n-1}, \quad n \in \mathbb{Z}, \quad \text{or}, \quad (2.38a)$$

$$y = \begin{bmatrix} \vdots \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ x_{-2} \\ \boxed{x_{-1}} \\ x_0 \\ \vdots \end{bmatrix} = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots \\ \ddots & 0 & 0 & 0 & \dots \\ \dots & 1 & \boxed{0} & 0 & \dots \\ \dots & 0 & 1 & 0 & \dots \\ \vdots & \vdots & \ddots & \ddots & \end{bmatrix} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ \vdots \end{bmatrix}. \quad (2.38b)$$

## 2.3. Systems

189

It is an LSI operator, causal and BIBO stable, but not memoryless; the matrix is Toeplitz, with a single nonzero off diagonal. A shift by  $k$ ,  $k > 0$ , is obtained by applying the delay operator  $k$  times.

The *advance-by-1* operator, which maps  $x_n$  into  $x_{n+1}$ , is the inverse of the shift-by-1 operator (2.38):

$$y_n = x_{n+1}, \quad n \in \mathbb{Z}, \quad \text{or,} \quad (2.39a)$$

$$y = \begin{bmatrix} \vdots \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ x_0 \\ \boxed{x_1} \\ x_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} \ddots & \ddots & \vdots & \vdots & \vdots \\ \cdots & 0 & 1 & 0 & \cdots \\ \cdots & 0 & \boxed{0} & 1 & \cdots \\ \cdots & 0 & 0 & 0 & \ddots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ \vdots \end{bmatrix}. \quad (2.39b)$$

It is an LSI operator, and BIBO stable, but neither memoryless nor causal; the matrix is Toeplitz and upper triangular with a single nonzero off diagonal. While it is obvious that the matrix in (2.39b) is the transpose of the one in (2.38b), it is also true that these matrices are inverses of each other. (Any finite-sized truncation of the matrix in (2.38b) or (2.39b), centered at the origin, is not invertible.)

**Modulator** Consider pointwise multiplication of a sequence  $x_n$  by  $(-1)^n$ :

$$y_n = (-1)^n x_n = \begin{cases} x_n, & \text{for even } n; \\ -x_n, & \text{for odd } n, \end{cases} \quad n \in \mathbb{Z}, \quad \text{or,} \quad (2.40a)$$

$$\begin{bmatrix} \vdots \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ -x_{-1} \\ \boxed{x_0} \\ -x_1 \\ \vdots \end{bmatrix} = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots \\ \cdots & -1 & 0 & 0 & \cdots \\ \cdots & 0 & \boxed{1} & 0 & \cdots \\ \cdots & 0 & 0 & -1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ \vdots \end{bmatrix}. \quad (2.40b)$$

This is the simplest example of *modulation*, that is, change of *frequency*<sup>38</sup> of the sequence. For example, a constant sequence  $x_n = 1$  turns into a fast-varying (high-frequency) sequence  $y_n = (-1)^n$ :

$$\left[ \dots \quad 1 \quad 1 \quad \boxed{1} \quad 1 \quad 1 \quad \dots \right]^T \rightarrow \left[ \dots \quad 1 \quad -1 \quad \boxed{1} \quad -1 \quad 1 \quad \dots \right]^T.$$

This operator is linear, causal, memoryless and BIBO stable, but not shift invariant; the matrix is diagonal.

A more general version of (2.40a) would involve a sequence  $\alpha_n$  multiplying

<sup>38</sup>While we have not defined the notion of frequency yet, you may think of it as a rate of variation in a sequence; the more the sequence varies in a given interval, the higher the frequency.

the input:

$$y_n = \alpha_n x_n, \quad n \in \mathbb{Z}, \quad \text{or,} \quad (2.41a)$$

$$\begin{bmatrix} \vdots \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ \alpha_{-1}x_{-1} \\ \boxed{\alpha_0 x_0} \\ \alpha_1 x_1 \\ \vdots \end{bmatrix} = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \\ \cdots & \alpha_{-1} & 0 & 0 & \cdots \\ \cdots & 0 & \boxed{\alpha_0} & 0 & \cdots \\ \cdots & 0 & 0 & \alpha_1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ \vdots \end{bmatrix}. \quad (2.41b)$$

Like (2.40), it is linear, causal, memoryless and BIBO stable, but not shift invariant; the matrix is again diagonal.

**Accumulator** The output of the accumulator is akin to the integral of the input:

$$y_n = \sum_{k=-\infty}^n x_k, \quad n \in \mathbb{Z}, \quad \text{or,} \quad (2.42a)$$

$$\begin{bmatrix} \vdots \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ \vdots \end{bmatrix} = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \\ \cdots & 1 & 0 & 0 & \cdots \\ \cdots & 1 & \boxed{1} & 0 & \cdots \\ \cdots & 1 & 1 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ \vdots \end{bmatrix}. \quad (2.42b)$$

This is an LSI, causal operator, but not memoryless nor BIBO stable; the matrix is Toeplitz and lower triangular.

If the input signal is restricted to be 0 for  $n < 0$ , (2.42) reduces to

$$y_n = \sum_{k=0}^n x_k, \quad n \in \mathbb{N}, \quad \text{or,} \quad (2.43a)$$

$$\begin{bmatrix} \boxed{y_0} \\ y_1 \\ y_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} \boxed{1} & 0 & 0 & \cdots \\ 1 & 1 & 0 & \cdots \\ 1 & 1 & 1 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \boxed{x_0} \\ x_1 \\ x_2 \\ \vdots \end{bmatrix}. \quad (2.43b)$$

This is an LSI, causal operator, but not memoryless nor BIBO stable; the matrix is Toeplitz and lower triangular.

Weighting by dividing (2.43a) by the number of terms involved turns the

## 2.3. Systems

191

accumulator into a running average:

$$y_n = \frac{1}{n+1} \sum_{k=0}^n x_k, \quad n \in \mathbb{N}, \quad \text{or}, \quad (2.44a)$$

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots \\ \frac{1}{2} & \frac{1}{2} & 0 & \cdots \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \end{bmatrix}. \quad (2.44b)$$

This is a linear operator, causal and BIBO stable, but not shift invariant nor memoryless; the matrix is lower triangular.

Other weight functions are possible, such as a decaying exponential weighting of the entries with factor  $\alpha \in (0, 1)$ :

$$y_n = \sum_{k=0}^n \alpha^{n-k} x_k, \quad n \in \mathbb{N}, \quad \text{or}, \quad (2.45a)$$

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots \\ \alpha & 1 & 0 & \cdots \\ \alpha^2 & \alpha & 1 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \end{bmatrix}. \quad (2.45b)$$

This is an LSI, causal operator, but not memoryless; the matrix is Toeplitz and lower triangular. It is BIBO stable because  $|\alpha| < 1$ .

**Averaging Operators** Consider a system that averages neighboring values, for example,

$$y_n = \frac{1}{3}(x_{n-1} + x_n + x_{n+1}), \quad n \in \mathbb{Z}, \quad \text{or}, \quad (2.46a)$$

$$\begin{bmatrix} \vdots \\ y_{-1} \\ y_0 \\ y_1 \\ \vdots \end{bmatrix} = \frac{1}{3} \begin{bmatrix} \ddots & \ddots & \ddots & \vdots \\ \ddots & 1 & 1 & 0 & \cdots \\ \ddots & 1 & 1 & 1 & \ddots \\ \cdots & 0 & 1 & 1 & \ddots \\ \vdots & \ddots & \ddots & \ddots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ x_{-1} \\ x_0 \\ x_1 \\ \vdots \end{bmatrix}. \quad (2.46b)$$

As we have seen in Example 2.2, this is a moving average filter with  $N = 3$ . It is called *moving average* since we look at the sequence through a window of size 3, compute the average value, and then move the window to compute the next average. This operator is LSI and BIBO stable, but neither memoryless nor causal; the matrix is Toeplitz.

We could obtain a causal version by simply delaying the moving average by  $(N - 1)/2$  samples in (2.5). For  $N = 3$  as here, this results in

$$y_n = \frac{1}{3}(x_{n-2} + x_{n-1} + x_n), \quad (2.47a)$$

$$\begin{bmatrix} \vdots \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ \vdots \end{bmatrix} = \frac{1}{3} \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \ddots \\ \dots & 1 & 0 & 0 & \dots \\ \dots & 1 & \boxed{1} & 0 & \dots \\ \dots & 1 & 1 & 1 & \dots \\ \ddots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ \vdots \end{bmatrix}, \quad (2.47b)$$

a delayed-by-1 version of (2.46). This operator is again LSI and BIBO stable but also causal, while still not memoryless; the matrix is Toeplitz and lower triangular.

An alternative is a *block average*,

$$y_n = \frac{1}{3}(x_{3n-1} + x_{3n} + x_{3n+1}), \quad (2.48a)$$

$$\begin{bmatrix} \vdots \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ \vdots \end{bmatrix} = \frac{1}{3} \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ \dots & 1 & \boxed{1} & 1 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & 1 & 1 & 1 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \end{bmatrix}. \quad (2.48b)$$

It is easy to see that (2.48a) is simply (2.46a) evaluated at multiples of 3. Similarly, the matrix in (2.48b) contains only every third row of the one in (2.46b). This is a linear and BIBO stable operator; it is neither shift invariant, nor memoryless, nor causal; the matrix is block diagonal.

A nonlinear version of the averaging operator could be

$$y_n = \text{median}([x_{n-1} \ x_n \ x_{n+1}]). \quad (2.49)$$

Instead of the average of the three terms, this operator takes the median value. This operator is shift invariant and BIBO stable but clearly neither linear, nor causal, nor memoryless.

**Maximum Operator** This simple operator computes the maximum value of the input up to the current time:

$$y_n = \max([\dots \ x_{n-2} \ x_{n-1} \ x_n]). \quad (2.50)$$

This operator is clearly neither linear nor memoryless, but it is causal, shift invariant, and BIBO stable.



## 2.3. Systems

193

			Linear Def. 2.1	Shift inv. Def. 2.4	Causal Def. 2.3	Memoryless Def. 2.2	BIBO stable Def. 2.5
Shift	(delay)	(2.38)	✓	✓	✓	×	✓
	advance	(2.39)	✓	✓	×	×	✓
Modulator		(2.40)	✓	×	✓	✓	✓
	general	(2.41)	✓	×	✓	✓	✓
Accumulator		(2.42)	✓	✓	✓	×	×
	restr. input	(2.43)	✓	✓	✓	×	×
	weighted	(2.44)	✓	×	✓	×	✓
	exp. weighted	(2.45)	✓	×	✓	×	✓ ( $ \alpha  < 1$ )
Averaging oper.		(2.46)	✓	✓	×	×	✓
	causal	(2.47)	✓	✓	✓	×	✓
	block	(2.48)	✓	×	×	×	✓
	median	(2.49)	×	✓	×	×	✓
Maximum oper.		(2.50)	×	✓	✓	×	✓
Matrix representation			✓	Toeplitz	Lower triangular	Diagonal	Rows absolutely summable

**Table 2.3:** Basic discrete-time systems and their properties. Matrix representation assumes linearity.

## 2.3.2 Difference Equations

An important class of discrete-time systems can be described by *linear difference equations* that relate the input sequence and past outputs to the current output,

$$y_n = \sum_{k \in \mathbb{Z}} b_k^{(n)} x_{n-k} - \sum_{k=1}^{\infty} a_k^{(n)} y_{n-k}. \quad (2.51)$$

If we require shift invariance, then the coefficients  $a_k^{(n)}$  and  $b_k^{(n)}$  are constant (do not depend on  $n$ ), and we get a *linear, constant-coefficient difference equation*,

$$y_n = \sum_{k \in \mathbb{Z}} b_k x_{n-k} - \sum_{k=1}^{\infty} a_k y_{n-k}. \quad (2.52)$$

Such an equation does not determine whether a system is causal. However, (2.52) is suggestive of a recursive computation of the output, forward in time (increasing  $n$ ); we will concentrate on such solutions. To make the system causal, we restrict the dependence on  $x$  to the current and past values, leading to

$$y_n = \sum_{k=0}^{\infty} b_k x_{n-k} - \sum_{k=1}^{\infty} a_k y_{n-k}. \quad (2.53)$$

Realizable systems will have only a finite number of nonzero coefficients  $a_k$ ,  $k \in \{1, 2, \dots, N\}$  and  $b_k$ ,  $k \in \{0, 1, \dots, M\}$ , reducing (2.53) to

$$y_n = \sum_{k=0}^M b_k x_{n-k} - \sum_{k=1}^N a_k y_{n-k}. \quad (2.54)$$

We discuss finding solutions to such difference equations in Appendix 2.A.2.

**EXAMPLE 2.9 (DIFFERENCE EQUATION OF THE ACCUMULATOR)** As an example, consider the accumulator seen in (2.42a):

$$y_n = \sum_{k=-\infty}^n x_k = x_n + \sum_{k=-\infty}^{n-1} x_k = x_n + y_{n-1}, \quad (2.55)$$

which is of the form (2.54), with  $b_0 = 1$ ,  $a_1 = -1$ . The infinite sum has been turned into a recursive formula (2.55), showing also how one could implement the accumulator: to obtain the current output  $y_n$ , add the current input  $x_n$  to the previously saved output  $y_{n-1}$ .

Let us take  $x_n = \delta_n$ , and see what the accumulator does. Assume we are given  $y_{-1} = \beta$ . Then, for  $n \geq 0$ ,

$$y_0 = x_0 + y_{-1} = 1 + \beta, \quad y_1 = x_1 + y_0 = 1 + \beta, \quad \dots, \quad y_n = 1 + \beta, \quad \dots$$

Thus, the accumulator does exactly what it is supposed to do: at time  $n = 0$ , it adds the value of the input  $x_0 = 1$  to the previously saved output  $y_{-1} = \beta$ , and then stays constant as the input for all  $n > 0$  is zero. For  $n < 0$ , we can solve (2.55) by expressing  $y_{n-1} = y_n - x_n$ ; it is easy to see that  $y_n = \beta$ , for all  $n < 0$ . Together, the expressions for  $n \geq 0$  and  $n < 0$ , lead to:

$$y_n = \beta + u_n, \quad (2.56)$$

that is, the initial value before the input is applied plus the input from the moment it is applied on.

From the above example, we see that unless the initial conditions are zero, the system is not linear, that is, one could have a zero input producing a nonzero output. Similarly, the system is shift invariant only if the initial conditions are zero. These properties are fundamental and hold beyond the case of the accumulator: difference equations as in (2.54) are linear and shift invariant if and only if initial conditions are zero. This also means that the homogeneous solution is necessarily zero (see Exercise 2.3).

### 2.3.3 Linear Shift-Invariant Systems

#### Impulse Response

A linear operator is specified by its outputs in response to each element of a basis for its domain space (see Section 1.5.5). As we saw in (2.33), this allows a matrix

representation of a linear discrete-time system to be formed from the output sequences in response to the Kronecker delta sequence and its shifts as inputs. When, in addition, the system is shift invariant, to satisfy (2.36) all these output sequences are themselves related by shifting. Thus, the system is specified completely by the output sequence resulting from the Kronecker delta sequence as the input.

**DEFINITION 2.6 (IMPULSE RESPONSE)** A sequence  $h$  is called the impulse response of LSI discrete-time system  $T$  when input  $\delta$  produces output  $h$ .

The impulse response  $h$  of a causal linear system always satisfies  $h_n = 0$  for all  $n < 0$ . This is required because, according to (2.35), the output in response to input  $\delta$  must match on  $\{-\infty, \dots, -2, -1\}$  to the  $\mathbf{0}$  output sequence that results from the  $\mathbf{0}$  input sequence.

**EXAMPLE 2.10 (IMPULSE RESPONSE FROM A DIFFERENCE EQUATION)** The linear, constant-coefficient difference equation (2.53) with zero initial conditions represents an LSI system. An impulse response of the system is an output that results from Kronecker delta input,  $x = \delta$ . Thus, an impulse response satisfies

$$h_n \stackrel{(a)}{=} \sum_{k=0}^{\infty} b_k \delta_{n-k} - \sum_{k=1}^{\infty} a_k h_{n-k} \stackrel{(b)}{=} b_n - \sum_{k=1}^{\infty} a_k h_{n-k}, \quad (2.57)$$

where (a) follows from (2.53); and (b) from the sifting property of the Kronecker delta sequence (see Table 2.1).

If we restrict our attention to causal systems, then the LCCDE uniquely specifies the system. The impulse response satisfies  $h_n = 0$  for all  $n < 0$ , and  $h_n$  can be computed for all  $n \geq 0$  by using (2.57) recursively for  $n = 0, 1, \dots$

### Convolution

The impulse response and its shifts form the columns of the matrix representation of an LSI system, as in (2.33). Expressing this as a summation is instructive and introduces the key concept of convolution.

Since a general input  $x$  to LSI system  $T$  can be written as  $x_n = \sum_{k \in \mathbb{Z}} x_k \delta_{n-k}$ , we can express the output as

$$y = Tx = T \sum_{k \in \mathbb{Z}} x_k \delta_{n-k} \stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} x_k T \delta_{n-k} \stackrel{(b)}{=} \sum_{k \in \mathbb{Z}} x_k h_{n-k} = h * x, \quad (2.58)$$

where (a) follows from linearity; and (b) from shift invariance and the definition of the impulse response, defining the *convolution*.<sup>39</sup>

<sup>39</sup>Convolution is sometimes called linear convolution to distinguish it from the circular convolution of finite-length sequences as in Definition 2.9.

**DEFINITION 2.7 (CONVOLUTION)** The convolution between sequences  $h$  and  $x$  is defined as

$$(Hx)_n = (h * x)_n = \sum_{k \in \mathbb{Z}} x_k h_{n-k} = \sum_{k \in \mathbb{Z}} x_{n-k} h_k, \quad (2.59)$$

where  $H$  is called the convolution operator associated with  $h$ .

When not clear from the context, we will use a subscript on the convolution operator  $*_n$  to denote the argument over which we perform the convolution (for example,  $x_{n-m} *_n h_{\ell-n} = \sum_k x_{k-m} h_{\ell-n+k}$ ).

**EXAMPLE 2.11 (SOLUTION TO THE LSI DIFFERENCE EQUATION OF THE ACCUMULATOR)**

Let us go back to the difference equation of the accumulator (2.55), and assume zero initial conditions,  $y_{-1} = 0$ . According to (2.57), the impulse response for this system (recall that  $b_0 = 1$  and  $a_0 = -1$ ), that is, a response to  $x_n = \delta_n$ , is

$$h_n = [\dots \ 0 \ \boxed{1} \ 1 \ 1 \ 1 \ \dots]^T. \quad (2.60a)$$

Similarly, the response of the system to  $x_n = \delta_{n-k}$  is  $h_{n-k}$ :

$$h_{n-k} = [\dots \ 0 \ \underbrace{\boxed{0} \ 0 \ \dots \ 0}_k \ 1 \ 1 \ 1 \ \dots]^T. \quad (2.60b)$$

By linearity, for a general input  $x$ , we can now use these to get the output using the convolution as in (2.59):

$$y = \begin{bmatrix} \vdots \\ \boxed{y_0} \\ y_1 \\ \vdots \\ y_n \\ \vdots \end{bmatrix} \stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} x_k \underbrace{\begin{bmatrix} \vdots \\ \boxed{0} \\ 0 \\ \vdots \\ 1 \\ \vdots \end{bmatrix}}_{h_{n-k}} = \underbrace{\begin{bmatrix} \vdots \\ \boxed{x_0} \\ x_0 \\ \vdots \\ x_0 \\ \vdots \end{bmatrix}}_{x_0 h_n} + \underbrace{\begin{bmatrix} \vdots \\ \boxed{0} \\ x_1 \\ \vdots \\ x_1 \\ \vdots \end{bmatrix}}_{x_1 h_{n-1}} + \dots + \underbrace{\begin{bmatrix} \vdots \\ \boxed{0} \\ 0 \\ \vdots \\ x_n \\ \vdots \end{bmatrix}}_{x_n h_{n-k}} + \dots = \begin{bmatrix} \vdots \\ \boxed{x_0} \\ x_0 + x_1 \\ \vdots \\ \sum_{k=0}^n x_k \\ \vdots \end{bmatrix},$$

where (a) follows both from the convolution expression (2.59), and (2.60). The above result thus performs exactly the accumulating function.

**Properties** The convolution (2.59) satisfies:

(i) *Connection to the inner product*

$$(h * x)_n = \sum_{k \in \mathbb{Z}} x_k h_{n-k} = \langle x_k, h_{n-k}^* \rangle_k. \quad (2.61a)$$

(ii) *Commutativity*

$$h * x = x * h. \quad (2.61b)$$

(iii) *Associativity*

$$g * (h * x) = g * h * x = (g * h) * x. \quad (2.61c)$$

(iv) *Deterministic autocorrelation*

$$a_n = \sum_{k \in \mathbb{Z}} x_k x_{k-n}^* = x_n * x_{-n}^*. \quad (2.61d)$$

All properties of convolution above depend on the sums—whether written explicitly or implicitly—converging. Convergence of the convolution is discussed in Appendix 2.A.3. The following example illustrates the apparent failure of the associative property when a convolution sum does not converge.

**EXAMPLE 2.12 (WHEN CONVOLUTION IS NOT ASSOCIATIVE)** Since convolutions may fail to converge, one needs to be careful about associativity. For  $g$ , choose the Heaviside sequence from (2.10),  $g_n = u_n$ , for  $h$  the first-order differencing sequence,  $h_n = \delta_n - \delta_{n-1}$ , and for  $x$  the constant sequence,  $x_n = 1$ . Now,

$$g * (h * x) \stackrel{(a)}{=} u * \mathbf{0} = \mathbf{0}, \quad \text{while} \quad (g * h) * x \stackrel{(b)}{=} \delta * 1 = 1,$$

where (a) follows because convolving a constant with the differencing operator yields a zero sequence; and (b) because convolving a Heaviside sequence with the differencing operator yields a Kronecker delta sequence. This failure of associativity occurs because  $g * h * x$  is not well defined; for it to be well defined requires absolute convergence of

$$\sum_{m \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} g_{n-m} h_{m-k} x_k$$

for every  $n \in \mathbb{Z}$ , which does not hold.

**Filters** The impulse response is often called a *filter* and the convolution is called *filtering*. Here are some basic classes of filters:

- (i) *Causal filters* are such that  $h_n = 0$  for all  $n < 0$ .
- (ii) *Anticausal filters* are such that  $h_n = 0$  for all  $n > 0$ .
- (iii) *Two-sided filters* are neither causal nor anticausal.
- (iv) *Finite impulse response (FIR) filters* have only a finite number of coefficients  $h_n$  different from zero.
- (v) *Infinite impulse response (IIR) filters* have an infinite number of nonzero terms.

For example, the impulse response in Example 2.11 is causal and IIR.

**Stability** We now discuss stability of LSI systems.

**PROPOSITION 2.8** An LSI system is BIBO stable if and only if its impulse response is absolutely summable.

*Proof.* To prove sufficiency (absolute summability implies BIBO stability), consider an absolutely-summable impulse response  $h \in \ell^1(\mathbb{Z})$ , so  $\|h\|_1 < \infty$ , and a bounded input  $x \in \ell^\infty(\mathbb{Z})$ , so  $\|x\|_\infty < \infty$ . The absolute value of any one sample at the output can be bounded as follows:

$$|y_n| \stackrel{(a)}{=} \left| \sum_{k \in \mathbb{Z}} h_k x_{n-k} \right| \stackrel{(b)}{\leq} \sum_{k \in \mathbb{Z}} |h_k| |x_{n-k}| \stackrel{(c)}{\leq} \|x\|_\infty \sum_{k \in \mathbb{Z}} |h_k| \stackrel{(d)}{=} \|x\|_\infty \|h\|_1 < \infty,$$

where (a) follows from (2.59); (b) from the triangle inequality (Definition 1.9(iii)); (c) from bounding each  $|x_{n-k}|$  by  $\|x\|_\infty$ ; and (d) from the definition of the  $\ell^1$  norm. This proves that  $y$  is bounded.<sup>40</sup>

We prove necessity (BIBO stability implies absolute summability) by contradiction. For any  $h$  that is not absolutely summable we choose a particular input  $x$  (which depends on  $h$ ) to create an unbounded output. Consider a real impulse response<sup>41</sup>  $h$ , and define the input sequence to be

$$x_n = \operatorname{sgn}(h_{-n}), \quad \text{where} \quad \operatorname{sgn}(t) = \begin{cases} -1, & \text{for } t < 0; \\ 0, & \text{for } t = 0; \\ 1, & \text{for } t > 0, \end{cases}$$

is the sign function. Now, compute the convolution of  $x$  with  $h$  at  $n = 0$ :

$$y_0 = \sum_{k \in \mathbb{Z}} h_k x_{-k} = \sum_{k \in \mathbb{Z}} |h_k| = \|h\|_1, \quad (2.62)$$

which is unbounded when  $h$  is not in  $\ell^1(\mathbb{Z})$ .

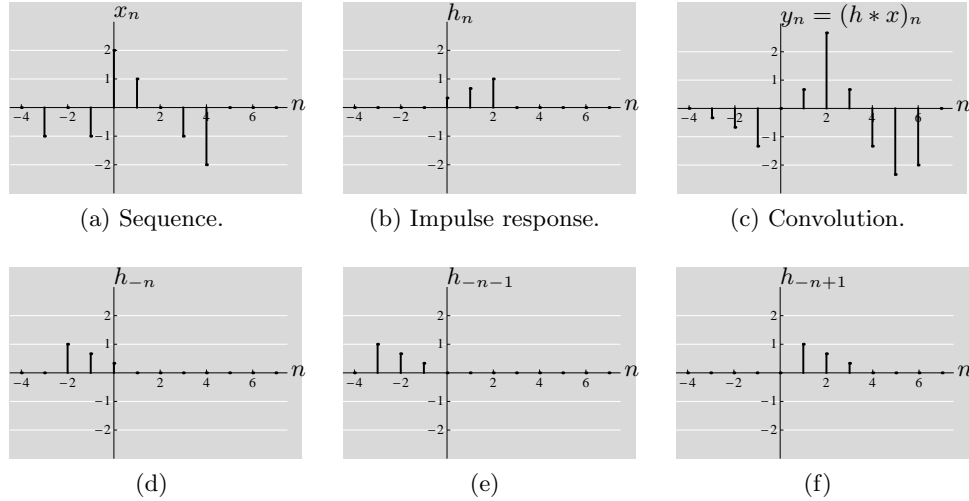
The impulse response of the accumulator, for example, does not belong to  $\ell^1(\mathbb{Z})$ ; a bounded input to the accumulator can lead to an unbounded output. Limiting attention to filters in  $\ell^1(\mathbb{Z})$  avoids technical difficulties with both convergence of the convolution sum as well as the resulting sequence being in a suitable sequence space. When  $h \in \ell^1(\mathbb{Z})$  and  $x \in \ell^p(\mathbb{Z})$  for any  $p \in [1, \infty]$ , the result of  $h * x$  is in  $\ell^p(\mathbb{Z})$  as well; see Solved Exercise 2.2.

<sup>40</sup>This boundedness is equivalent to the convergence of the convolution sum as discussed in Appendix 2.A.3.

<sup>41</sup>For a complex-valued impulse response, a slight modification, using  $x_n = h_n^* / |h_n|$  for  $|h_n| \neq 0$ , and  $x_n = 0$  otherwise, leads to the same result.

## 2.3. Systems

199



**Figure 2.5:** Example of the convolution between a sequence and a filter. (a) Sequence  $x_n$ . (b) Impulse response  $h_n$ . (c) Result of convolution  $y_n$ . (d) Time-reversed version of the impulse response,  $h_{-n}$ . (e)-(f) Two time-reversed and shifted versions of the impulse response involved in computing the convolution.

**Matrix View** As we have shown in Section 1.5.5, any linear operator can be expressed in matrix form. We may visualize (2.59) as:

$$y = \begin{bmatrix} \vdots \\ y_{-2} \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ y_2 \\ \vdots \end{bmatrix} = \underbrace{\begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & h_0 & h_{-1} & h_{-2} & h_{-3} & h_{-4} & \dots \\ \dots & h_1 & h_0 & h_{-1} & h_{-2} & h_{-3} & \dots \\ \dots & h_2 & h_1 & \boxed{h_0} & h_{-1} & h_{-2} & \dots \\ \dots & h_3 & h_2 & h_1 & h_0 & h_{-1} & \dots \\ \dots & h_4 & h_3 & h_2 & h_1 & h_0 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_H \begin{bmatrix} \vdots \\ x_{-2} \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ x_2 \\ \vdots \end{bmatrix} = Hx. \quad (2.63)$$

This again shows that an LSI discrete-time system, linear operator (on sequences), filter and (doubly-infinite) matrix are all synonyms. The key elements in (2.63) are the time reversal of the impulse response (in each row of the matrix, the impulse response goes from right to left), and the Toeplitz structure of the matrix (each row is a shifted version of the previous row, the matrix is constant along diagonals, see (1.228)). In Figure 2.5, an example convolution is computed graphically, emphasizing time reversal.

We can easily find the matrix form of the adjoint of the convolution operator, as the unique  $H^*$  satisfying (1.44):

$$\langle Hx, y \rangle = \langle x, H^*y \rangle. \quad (2.64)$$

The result is the convolution operator associated with  $h_{-n}^*$ , the time-reversed and conjugated version of  $h_n$ :

$$H^* = \begin{bmatrix} \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ \ddots & h_0^* & h_1^* & h_2^* & h_3^* & h_4^* \\ \ddots & h_{-1}^* & h_0^* & h_1^* & h_2^* & h_3^* & \ddots \\ \ddots & h_{-2}^* & h_{-1}^* & \boxed{h_0^*} & h_1^* & h_2^* & \ddots \\ \ddots & h_{-3}^* & h_{-2}^* & h_{-1}^* & h_0^* & h_1^* & \ddots \\ & h_{-4}^* & h_{-3}^* & h_{-2}^* & h_{-1}^* & h_0^* & \ddots \\ \ddots & & \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}. \quad (2.65)$$

### Circular Convolution

We now consider what happens with our second class of sequences, those that are of finite length circularly extended. To start, we assume that the impulse response  $h$  is in  $\ell^1(\mathbb{Z})$ .

**Linear Convolution with Circularly-Extended Signal** Given a sequence  $x$  with circular extension as in (2.25) and a filter  $h$  in  $\ell^1(\mathbb{Z})$ , we can compute the convolution as usual:

$$y_n = (h * x)_n = \sum_{k \in \mathbb{Z}} x_k h_{n-k} = \sum_{k \in \mathbb{Z}} h_k x_{n-k}. \quad (2.66)$$

Since  $x$  is  $N$ -periodic,  $y$  is  $N$ -periodic as well:

$$y_{n+N} = \sum_{k \in \mathbb{Z}} h_k x_{n+N-k} \stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} h_k x_{n-k} = y_n,$$

where (a) follows from the periodicity of  $x$ .

Let us now define a periodized version of  $h$ , with period  $N$ , as:

$$h_{N,n} = \sum_{k \in \mathbb{Z}} h_{n-kN}, \quad (2.67)$$

which converges for every  $n$  because  $h \in \ell^1(\mathbb{Z})$ . We now want to show how we can



express the convolution (2.66)<sup>42</sup> in terms of what we define as a *circular* convolution:

$$\begin{aligned}
 (h * x)_n &= \sum_{k \in \mathbb{Z}} h_k x_{n-k} \stackrel{(a)}{=} \sum_{\ell \in \mathbb{Z}} \sum_{k=\ell N}^{(\ell+1)N-1} h_k x_{n-k} \\
 &\stackrel{(b)}{=} \sum_{\ell \in \mathbb{Z}} \sum_{k'=0}^{N-1} h_{k'+\ell N} x_{n-k'-\ell N} \stackrel{(c)}{=} \sum_{\ell \in \mathbb{Z}} \sum_{k=0}^{N-1} h_{k+\ell N} x_{n-k} \\
 &\stackrel{(d)}{=} \sum_{k=0}^{N-1} \underbrace{\sum_{\ell \in \mathbb{Z}} h_{k+\ell N}}_{h_{N,k}} x_{n-k} = \sum_{k=0}^{N-1} h_{N,k} x_{n-k} \\
 &\stackrel{(e)}{=} \sum_{k=0}^{N-1} h_{N,k} x_{(n-k) \bmod N} = (h_N \circledast x)_n, \tag{2.68}
 \end{aligned}$$

where in (a) we split the set of integers into length- $N$  segments; (b) follows from change of variable  $k' = k - \ell N$ ; (c) follows from periodicity of  $x$  and change of variable  $k = k'$ ; in (d) we were allowed to exchange order of summation because  $h \in \ell^1(\mathbb{Z})$  (see Appendix 1.A.3); and (e) follows from periodicity of  $x$ . The expression above tends to be more convenient as it involves only one period of both  $x$  and the periodized version  $h_N$  of the impulse response  $h$ . The relationships between the convolution of a finite-length sequence  $x$  circularly extended, and  $h$ , and the circular convolution of the same sequence  $x$ , and  $h_N$  are shown in Figure 2.6 and explored further in the context of circulant matrices in Solved Exercise 2.5.

**Definition of the Circular Convolution** Above in (2.68), we implicitly defined a new form of convolution between a length- $N$  input sequence  $x$  and a length- $N$  impulse response  $h$ :

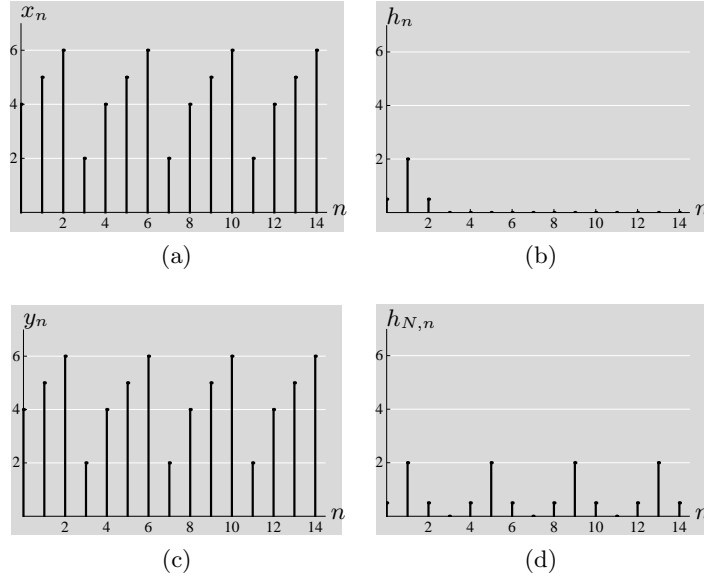
**DEFINITION 2.9 (CIRCULAR CONVOLUTION)** The circular convolution between length- $N$  sequences  $h$  and  $x$  is defined as

$$(Hx)_n = (h \circledast x)_n = \sum_{k=0}^{N-1} x_k h_{(n-k) \bmod N} = \sum_{k=0}^{N-1} x_{(n-k) \bmod N} h_k, \tag{2.69}$$

where  $H$  is called the circular convolution operator associated with  $h$ .

The result of the circular convolution is a length- $N$  sequence. While this notion of convolution is independent from that of linear convolution, we have just seen that the two are related when the input sequence is of finite length (circularly extended) but the impulse response of the system is not. We made the connection by periodizing that impulse response.

<sup>42</sup>When there is more than one form of convolution involved, we term the one in (2.66) *linear* convolution.



**Figure 2.6:** Convolution of (a) finite-length sequence  $x$ , circularly extended (periodic sequence of period  $N = 4$ ). (b) The filter  $h$ , (c) convolved with the sequence  $x$ , leads to a finite-length output  $y$ , circularly extended. (d) The equivalent, periodized filter  $h_N = h_4$ , circularly convolved with  $x$ , leads to the same output as in (c).

**Equivalence of Circular and Linear Convolutions** While we have seen that linear and circular convolutions are related, there are instances when the two are equivalent. Assume we have a length- $M$  input  $x$  and a length- $L$  impulse response  $h$ :

$$\begin{aligned} x &= [\dots \ 0 \ \boxed{x_0} \ x_1 \ \dots \ \dots \ x_{M-1} \ 0 \ \dots]^T, \\ h &= [\dots \ 0 \ \boxed{h_0} \ h_1 \ \dots \ h_{L-1} \ 0 \ \dots]^T. \end{aligned} \quad (2.70)$$

The result of the linear convolution (2.59) has at most  $L + M - 1$  nonzero samples:

$$y = [\dots \ 0 \ \boxed{y_0} \ y_1 \ \dots \ y_{L+M-1} \ 0 \ \dots]^T.$$

While we have chosen to write the sequences as infinite-length vectors, we could have also chosen to write each as a finite-length vector with appropriate length; however, as these lengths are all different, we would have had to choose a common vector length  $N$ . Choosing this common length is exactly the crucial point in when the linear and circular convolutions are equivalent, as we show next.

**PROPOSITION 2.10 (EQUIVALENCE OF CIRCULAR AND LINEAR CONVOLUTIONS)**  
Linear and circular convolutions between a length- $M$  sequence  $x$  and a length- $L$

sequence  $h$  are equivalent when the period of the circular convolution  $N$  satisfies

$$N \geq L + M - 1. \quad (2.71)$$

*Proof.* Take  $x$  and  $h$  as in (2.70). The linear and circular convolutions,  $y^{(\text{lin})}$  and  $y^{(\text{circ})}$ , are given by (2.59) and (2.69), respectively:

$$y_n^{(\text{lin})} = (h_0 x_n + \dots + h_n x_0) + (h_{n+1} x_{-1} + \dots + h_{L-1} x_{-L+1+n}), \quad (2.72a)$$

$$y_n^{(\text{circ})} = (h_0 x_n + \dots + h_n x_0) + (h_{n+1} x_{N-1} + \dots + h_{L-1} x_{N-L+1+n}), \quad (2.72b)$$

for  $n \in \{0, 1, \dots, N-1\}$ . In the above, we broke each convolution sum into positive indices of  $x_n$  and the rest (negative ones for the linear convolution, and mod  $N$  for the circular convolution). Note that in (2.72b) the index goes from 0 to  $(N-1)$ , but stops at  $(L-1)$  since  $h$  is zero after that.

Since  $x$  has no nonzero values for negative values of  $n$ , the second sum in (2.72a) is zero, and so must the second sum in (2.72b) be (and that for every  $n = 0, 1, \dots, N-1$ ), if (2.72a) and (2.72b) are to be equal. This, in turn, is possible only if  $x_{N-L+1+n}$  (the last  $x$  term in the second sum of the circular convolution) has an index that is outside the range of nonzero values of  $x$ , that is, if  $N - L + 1 + n \geq M$ , for every  $n = 0, 1, \dots, N-1$ . As this is true for  $n = 0$  by assumption (2.71), it will be true for all larger  $n$  as well.

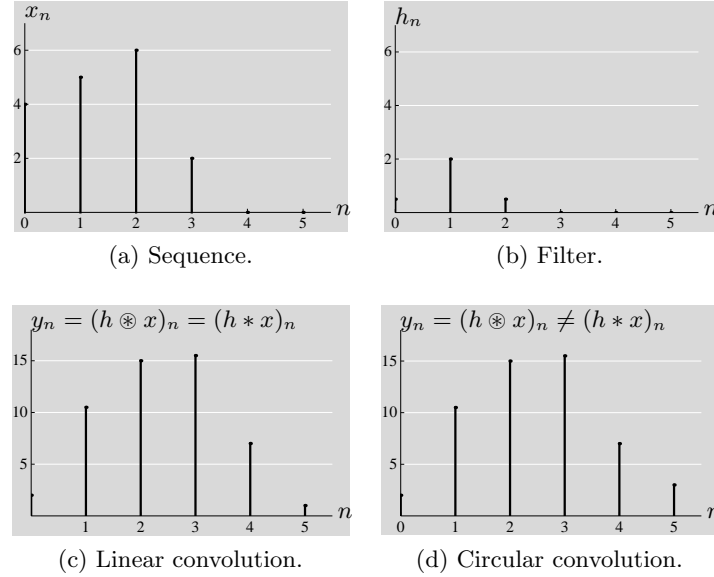
Figure 2.7 depicts this equivalence and Example 2.13 examines it in matrix notation for  $M = 4$ ,  $L = 3$ .

**Matrix View** As we have done for linear convolution in (2.63), we visualize circular convolution (2.69) using matrices:

$$y = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_{N-1} \end{bmatrix} = \underbrace{\begin{bmatrix} h_0 & h_{N-1} & h_{N-2} & \dots & h_1 \\ h_1 & h_0 & h_{N-1} & \dots & h_2 \\ h_2 & h_1 & h_0 & \dots & h_3 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ h_{N-1} & h_{N-2} & h_{N-3} & \dots & h_0 \end{bmatrix}}_H \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{N-1} \end{bmatrix} = Hx. \quad (2.73)$$

$H$  is a circulant matrix as in (1.227) with  $h$  as its first column, and it represents the circular convolution operator when both the sequence  $x$  and impulse response  $h$  are finite; when the impulse response is not finite, the elements of  $H$  would be samples of the periodized impulse response  $h_N$ .

**EXAMPLE 2.13 (EQUIVALENCE OF CIRCULAR AND LINEAR CONVOLUTIONS)** We now look at a length-3 filter convolved with a length-4 sequence. The result of



**Figure 2.7:** Equivalence of circular and linear convolutions. (a) Sequence  $x$  of length  $M = 4$ . (b) Filter  $h$  of length  $L = 3$ . (c) Linear convolution results in a sequence of length  $L + M - 1 = 6$ , the same as a circular convolution with a period  $N \geq L + M - 1$ ,  $N = 6$  in this case. (d) However, circular convolution with a smaller period,  $N = 5$ , does not lead to the same result.

the linear convolution is of length 6:

$$\begin{bmatrix} \vdots \\ 0 \\ \boxed{y_0} \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ 0 \\ \vdots \end{bmatrix} = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & h_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ \dots & h_1 & \boxed{h_0} & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ \dots & h_2 & h_1 & h_0 & 0 & 0 & 0 & 0 & 0 & \dots \\ \dots & 0 & h_2 & h_1 & h_0 & 0 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & h_2 & h_1 & h_0 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & h_2 & h_1 & h_0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & 0 & h_2 & h_1 & h_0 & 0 & \dots \\ \dots & 0 & 0 & 0 & 0 & 0 & h_2 & h_1 & h_0 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ 0 \\ \boxed{x_0} \\ x_1 \\ x_2 \\ x_3 \\ 0 \\ 0 \\ 0 \\ \vdots \end{bmatrix}. \quad (2.74a)$$

To calculate circular convolution, we choose  $N = M + L - 1 = 6$ , and form a  $6 \times 6$  circulant matrix  $H$  as in (2.73) by using  $h$  as its first column. Then the

circular convolution leads to the same result as before:

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} h_0 & 0 & 0 & 0 & h_2 & h_1 \\ h_1 & h_0 & 0 & 0 & 0 & h_2 \\ h_2 & h_1 & h_0 & 0 & 0 & 0 \\ 0 & h_2 & h_1 & h_0 & 0 & 0 \\ 0 & 0 & h_2 & h_1 & h_0 & 0 \\ 0 & 0 & 0 & h_2 & h_1 & h_0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ 0 \\ 0 \end{bmatrix}. \quad (2.74b)$$

Had the period  $N$  been chosen smaller (for example,  $N = 5$ ), the equivalence would have not held.

This example also shows that to compute the linear convolution, we can compute the circular convolution instead by choosing the appropriate period  $N \geq M + L - 1$ . This is often done as the circular convolution can be computed using the discrete Fourier transform of size  $N$  (see Section 2.9.2), and fast algorithms for the discrete Fourier transform abound (see Section 2.9.1).

## 2.4 Discrete-Time Fourier Transform

In this and the next two sections, we introduce various ways to analyze sequences and discrete-time systems. They range from the analytical to the computational and are all variations of the Fourier transform. Why this prominent role of Fourier methods? Simply because they are based on eigensequences of LSI systems (convolution operators). Thus far, we have seen two convolution operators (linear and circular). We will see that these have different sets of eigensequences, which lead to different Fourier transforms for sequences. The eigensequence property leads to the diagonalization of the convolution operator, which then implies the *convolution property*—an equivalence between convolving sequences and multiplying Fourier transforms of the sequences.

In this section, we introduce the *discrete-time Fourier transform (DTFT)*—the Fourier transform for infinite-length discrete-time sequences. It is a  $2\pi$ -periodic function of frequency  $\omega$  that we write as  $X(e^{j\omega})$ , with  $e^{j\omega}$  clearly in  $\mathbb{C}$  and of modulus 1, both to stress periodicity as well as to create a unified notation for the DTFT and the  *$z$ -transform*  $X(z)$ , which we discuss in Section 2.5. The  $z$ -transform has argument  $z \in \mathbb{C}$  that may have modulus different from 1. In Section 2.6, we focus on the *discrete Fourier transform (DFT)*—the Fourier transform for both infinite-length periodic sequences as well as length- $N$  sequences circularly extended (both of these can be viewed as existing on a discrete circle of length  $N$ ). The DFT is an  $N$ -dimensional vector we write as  $X_k$ .

### 2.4.1 Definition of the DTFT

**Eigensequences of the Convolution Operator** We start with a fundamental property of LSI systems: they all have all unit-modulus complex exponential sequences as eigensequences. This follows from the convolution representation of LSI systems (2.59) and a simple computation.

Consider a complex exponential sequence

$$v_n = e^{j\omega n}, \quad n \in \mathbb{Z}, \quad (2.75)$$

where  $\omega$  is any real number. The quantity  $\omega$  is called *angular frequency*; it is measured in radians per second. With  $\omega = 2\pi f$ , the quantity  $f$  is called *frequency*; it is measured in Hertz, or the number of cycles per second. The sequence  $v$  is bounded since  $|v_n| = 1$  for all  $n \in \mathbb{Z}$ . If the impulse response  $h$  is in  $\ell^1(\mathbb{Z})$ , according to Proposition 2.8, the output  $h * v$  is bounded as well. Along with being bounded,  $h * v$  takes a particular form:

$$\begin{aligned} (Hv)_n &= (h * v)_n = \sum_{k \in \mathbb{Z}} v_{n-k} h_k = \sum_{k \in \mathbb{Z}} e^{j\omega(n-k)} h_k \\ &= \underbrace{\sum_{k \in \mathbb{Z}} h_k e^{-j\omega k}}_{\lambda_\omega} \underbrace{e^{j\omega n}}_{v_n}. \end{aligned} \quad (2.76)$$

This shows that applying the convolution operator  $H$  to the complex exponential sequence  $v$  gives a scalar multiple of  $v$ . In other words,  $v$  is an eigensequence of  $H$  with corresponding eigenvalue  $\lambda_\omega$  that we call the *frequency response* of the system  $H(e^{j\omega})$ ; it is defined formally in (2.105a). We can thus rewrite (2.76) as

$$H e^{j\omega n} = h * e^{j\omega n} = H(e^{j\omega}) e^{j\omega n}. \quad (2.77)$$

The scalar multiples of  $v$  form a subspace  $S_\omega = \{\alpha e^{j\omega n} \mid \alpha \in \mathbb{C}\}$ . This space is invariant under the operation of convolution: when the input is in  $S_\omega$ , the output is in  $S_\omega$  as well.

**DTFT** Finding the appropriate Fourier transform now amounts to projecting onto each of the invariant subspaces  $S_\omega$ .

**DEFINITION 2.11 (DISCRETE-TIME FOURIER TRANSFORM)** The discrete-time Fourier transform of a sequence  $x$  is

$$X(e^{j\omega}) = \sum_{n \in \mathbb{Z}} x_n e^{-j\omega n}, \quad \omega \in \mathbb{R}. \quad (2.78a)$$

It exists when (2.78a) converges for all  $\omega \in \mathbb{R}$ ; we then call it the *spectrum* of  $x$ . The inverse DTFT of a  $2\pi$ -periodic function  $X(e^{j\omega})$  is

$$x_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega, \quad n \in \mathbb{Z}. \quad (2.78b)$$

When the DTFT exists, we denote the DTFT pair as

$$x_n \xleftrightarrow{\text{DTFT}} X(e^{j\omega}).$$

Since  $e^{-j\omega n}$  is a  $2\pi$ -periodic function of  $\omega$  for every  $n \in \mathbb{Z}$ , the DTFT is always a  $2\pi$ -periodic function, which is emphasized by the notation  $X(e^{j\omega})$ . Note that the sum in (2.78a) is formally equivalent to an  $\ell^2(\mathbb{Z})$  inner product, although the sequence  $e^{j\omega n}$  has no decay and is thus not in  $\ell^2(\mathbb{Z})$ . We now discuss limitations on the inputs and the corresponding types of convergence.

### 2.4.2 Existence and Convergence of the DTFT

The existence of the DTFT depends on the sequence  $x$ . When a doubly-infinite series as in (2.78a) is given without a specification of how to interpret it as a limiting process, one must consider the series well defined only when it converges absolutely (see Appendix 1.A.2). This immediately implies existence of the DTFT for all sequences in  $\ell^1(\mathbb{Z})$ . To extend beyond  $\ell^1(\mathbb{Z})$ , we consider the limit as  $N \rightarrow \infty$  of the partial sums

$$X_N(e^{j\omega}) = \sum_{n=-N}^N x_n e^{-j\omega n}. \quad (2.79)$$

Convergence of the partial sums under the  $\mathcal{L}^2([-\pi, \pi])$  norm allows us to consider the DTFT to exist for all sequences in  $\ell^2(\mathbb{Z})$ . The DTFT can be a useful tool even when (2.78a) diverges to  $\infty$  for some values of  $\omega$ ; this requires more caution.

**Sequences in  $\ell^1(\mathbb{Z})$**  If  $x \in \ell^1(\mathbb{Z})$ , then (2.78a) converges absolutely for every  $\omega$ , since

$$\sum_{n \in \mathbb{Z}} |x_n e^{j\omega n}| = \sum_{n \in \mathbb{Z}} |x_n| |e^{j\omega n}| = \|x\|_1 < \infty.$$

This tells us that the DTFT of  $x$  exists. Moreover, as a consequence of absolute convergence for all  $\omega$ , the limit  $X(e^{j\omega})$  is a continuous function of  $\omega$ .<sup>43</sup>

Since the DTFT itself is well defined, we can verify the inversion formula by substituting (2.78a) into (2.78b). First,

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} \left( \sum_{k \in \mathbb{Z}} x_k e^{-j\omega k} \right) e^{j\omega n} d\omega \stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} x_k \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j\omega(n-k)} d\omega, \quad (2.80a)$$

where in (a) we are allowed to exchange the order of summation and integration because  $x \in \ell^1(\mathbb{Z})$  (see Section 1.A.3). The integral  $\int_{-\pi}^{\pi} e^{j\omega(n-k)} d\omega$  must be treated separately for  $n = k$  and  $n \neq k$ . Each case gives an elementary computation, and the result is

$$\int_{-\pi}^{\pi} e^{j\omega(n-k)} d\omega = 2\pi \delta_{n-k} \quad (2.80b)$$

<sup>43</sup>Absolute convergence of (2.78a) implies uniform convergence of the sequence of functions  $X_N$  in (2.79) to  $X$ . Looking at the DTFT as a function defined on a compact (closed and bounded) domain such as  $[-\pi, \pi]$ , the uniform convergence and the continuity of each  $X_N$  implies  $X$  is continuous.

using the Kronecker delta sequence to combine the cases. We can then rewrite (2.80a) as

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} \left( \sum_{k \in \mathbb{Z}} x_k e^{-j\omega k} \right) e^{j\omega n} d\omega \stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} x_k \delta_{n-k} \stackrel{(b)}{=} x_n,$$

where (a) follows from (2.80b); and (b) from the definition of the Kronecker delta sequence (2.7), proving the inversion.

**Sequences in  $\ell^2(\mathbb{Z})$**  For sequences not in  $\ell^1(\mathbb{Z})$ , the DTFT series (2.78a) may fail to converge for some values of  $\omega$ . Nevertheless, convergence can be extended to the larger space of sequences  $\ell^2(\mathbb{Z})$  by changing the sense of convergence.

If  $x \in \ell^2(\mathbb{Z})$ , the partial sum  $X_N(e^{j\omega})$  in (2.79) converges to a function  $X(e^{j\omega}) \in \mathcal{L}^2([-\pi, \pi])$  in the sense that

$$\lim_{N \rightarrow \infty} \|X(e^{j\omega}) - X_N(e^{j\omega})\| = 0. \quad (2.81)$$

This convergence in  $\mathcal{L}^2([-\pi, \pi])$  norm<sup>44</sup> implies convergence of (2.78a) for almost all values of  $\omega$ , but there is no guarantee of the convergence being uniform or the limit function  $X(e^{j\omega})$  being continuous.

The sense in which the inversion formula holds changes subtly as well. We return to this in Section 3.5.1.

**EXAMPLE 2.14 (MEAN-SQUARE CONVERGENCE OF DTFT)** Take the sinc sequence from Figure 2.1(b),

$$x_n = \frac{1}{\sqrt{2}} \text{sinc}(\pi n/2) = \frac{1}{\sqrt{2}} \frac{\sin(\pi n/2)}{\pi n/2}. \quad (2.82)$$

It decays too slowly to be absolutely summable but fast enough to be square summable; that is,  $x \in \ell^2(\mathbb{Z})$  but  $x \notin \ell^1(\mathbb{Z})$ . Thus, we cannot guarantee that (2.78a) converges for every  $\omega$ , but the DTFT still converges in mean square. To see this, in Figure 2.8 we plot the DTFT partial sum (2.79),

$$X_N(e^{j\omega}) = \frac{1}{\sqrt{2}} \sum_{n=-N}^N \frac{\sin(\pi n/2)}{\pi n/2} e^{-j\omega n}, \quad (2.83)$$

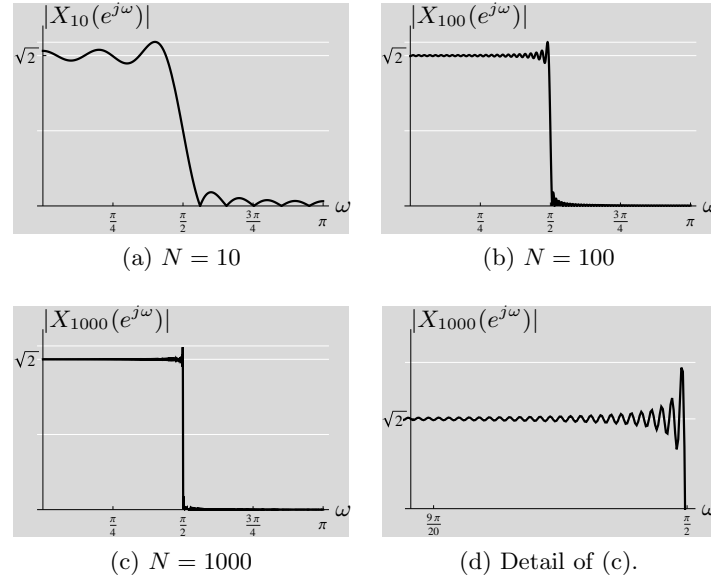
for various values of  $N$ . As the figure suggests, convergence in mean square is to

$$X(e^{j\omega}) = \begin{cases} \sqrt{2}, & \text{for } |\omega| < \pi/2; \\ 0, & \text{for } |\omega| \in (\pi/2, \pi], \end{cases}$$

and the convergence as  $N \rightarrow \infty$  is nonuniform: it is very slow near  $\omega = \pi/2$  and faster farther away. In fact, while there is no convergence at  $\omega = \pi/2$ , lack of convergence at this isolated point does not prevent convergence in mean square.

<sup>44</sup>This is also called *convergence in the mean-square sense*.





**Figure 2.8:** Truncated DTFT of the sinc sequence, illustrating the Gibbs phenomenon. Shown are  $|X_N(e^{j\omega})|$  from (2.83) with different  $N$ . Observe how oscillations narrow from (a) to (c), but their amplitude remains constant (the topmost grid line in every plot),  $1.089\sqrt{2}$ .

The partial sum  $X(e^{j\omega})$  oscillates near the points of discontinuity, with oscillations becoming narrower as  $N$  increases but not decreasing in size. This over- and undershoot is of the order of 9%, and is called the *Gibbs phenomenon* (see also Figure 0.3).<sup>45</sup>

**Using the DTFT Without Convergence** The DTFT is still a useful tool even in some cases where it converges neither pointwise over  $\omega$  nor in mean square. These are cases where an expression for the DTFT involving a Dirac delta function makes sense because evaluating the inverse DTFT gives the desired result. As with other uses of the Dirac delta function, we must be cautious.

Consider the constant sequence  $x_n = 1$  for all  $n \in \mathbb{Z}$ . This sequence belongs to neither  $\ell^1(\mathbb{Z})$  nor  $\ell^2(\mathbb{Z})$ , so neither of our previous discussions of convergence apply. In fact, there is no value of  $\omega$  for which the DTFT series (2.78a) converges. However, the lack of convergence is not the same for all values of  $\omega$ . When  $\omega$  is an integer multiple of  $2\pi$ , (2.78a) diverges to  $\infty$  because every term in the sum is 1. For other values of  $\omega$ , it is tempting (but *not* mathematically correct; see Appendix 1.A.2) to assign the value of zero to the sum because the terms in (2.78a) all lie on the unit circle, with no direction preferred. This gives some intuition for

<sup>45</sup>For any piecewise continuously differentiable function with a discontinuity of height  $\alpha$ , the overshoot is  $0.089\alpha$ , roughly 9% higher than the original height.

considering the DTFT to be

$$X(e^{j\omega}) = \begin{cases} \infty, & \text{for } \omega = 0; \\ 0, & \text{otherwise} \end{cases}$$

on  $[-\pi, \pi]$ . This mathematically nonsensical statement can be replaced by the dangerous but useful statement

$$X(e^{j\omega}) = 2\pi \delta(\omega) \quad \text{for } \omega \in [-\pi, \pi].$$

Substituting this in the inverse DTFT (2.78b) recovers the sequence  $x_n = 1$  for all  $n \in \mathbb{Z}$  that we started with. A very similar argument supports assigning the DTFT of  $2\pi \delta(\omega - \omega_0)$  to the complex exponential sequence  $e^{j\omega_0 n}$  for any  $\omega_0 \in (-\pi, \pi)$ .

### 2.4.3 Properties of the DTFT

We list here the basic properties of the DTFT; Table 2.4 summarizes these, together with symmetries as well as a few standard transform pairs. Of course, all the expressions must be well defined for these properties to hold.

**Linearity** The DTFT operator  $F$  is a linear operator, or,

$$\alpha x_n + \beta y_n \xleftrightarrow{\text{DTFT}} \alpha X(e^{j\omega}) + \beta Y(e^{j\omega}). \quad (2.84)$$

**Shift in Time** The DTFT pair corresponding to a shift in time by  $n_0$  is

$$x_{n-n_0} \xleftrightarrow{\text{DTFT}} e^{-j\omega n_0} X(e^{j\omega}). \quad (2.85)$$

**Shift in Frequency** The DTFT pair corresponding to a shift in frequency by  $\omega_0$  is

$$e^{j\omega_0 n} x_n \xleftrightarrow{\text{DTFT}} X(e^{j(\omega-\omega_0)}). \quad (2.86)$$

The shift in time and shift in frequency are the first of several Fourier transform properties that are *duals* in that swapping the roles of time and frequency results in a pair of similar statements.<sup>46</sup> Shifting in time is equivalent to modulation in frequency, and shifting in frequency is equivalent to modulation in time.

**Scaling in Time** Scaling in time appears in two flavors:

- (i) The DTFT pair corresponding to scaling in time by  $N$  is

$$x_{Nn} \xleftrightarrow{\text{DTFT}} \frac{1}{N} \sum_{k=0}^{N-1} X(e^{j(\omega-2\pi k)/N}). \quad (2.87)$$

This type of scaling is referred to as *downsampling*; we will discuss it in more detail in Section 2.7.

<sup>46</sup>In this section, time is discrete and frequency is continuous; dualities are more transparent when both are discrete (see Section 2.6) or both are continuous (see Section 3.4).

(ii) The DTFT pair corresponding to scaling in time by  $1/N$  is

$$\begin{cases} x_{n/N}, & \text{for } n = \ell N, \quad \ell \in \mathbb{Z}; \\ 0, & \text{otherwise,} \end{cases} \quad \stackrel{\text{DTFT}}{\longleftrightarrow} \quad X(e^{jN\omega}). \quad (2.88)$$

This type of scaling is referred to as *upsampling*; we will discuss it in more detail in Section 2.7.

**Time Reversal** The DTFT pair corresponding to a time reversal  $x_{-n}$  is

$$x_{-n} \quad \stackrel{\text{DTFT}}{\longleftrightarrow} \quad X(e^{-j\omega}). \quad (2.89)$$

For a real  $x_n$ , the DTFT of the time-reversed version  $x_{-n}$  is  $X^*(e^{j\omega})$ .

**Differentiation** The DTFT pair corresponding to differentiation in frequency is

$$(-jn)^k x_n \quad \stackrel{\text{DTFT}}{\longleftrightarrow} \quad \frac{\partial^k X(e^{j\omega})}{\partial \omega^k}. \quad (2.90)$$

**Moments** Computing the  $n$ th moment using the DTFT results in

$$m_k = \sum_{n \in \mathbb{Z}} n^k x_n = \left( \sum_{n \in \mathbb{Z}} n^k x_n e^{-j\omega n} \right) \Big|_{\omega=0} = (-j)^k \frac{\partial^k X(e^{j\omega})}{\partial \omega^k} \Big|_{\omega=0}, \quad k \in \mathbb{N}, \quad (2.91a)$$

as a direct application of (2.90). The first two moments are:

$$m_0 = \sum_{n \in \mathbb{Z}} x_n = \left( \sum_{n \in \mathbb{Z}} x_n e^{-j\omega n} \right) \Big|_{\omega=0} = X(0), \quad (2.91b)$$

$$m_1 = \sum_{n \in \mathbb{Z}} n x_n = \left( \sum_{n \in \mathbb{Z}} n x_n e^{-j\omega n} \right) \Big|_{\omega=0} = -j \frac{\partial X(e^{j\omega})}{\partial \omega} \Big|_{\omega=0}. \quad (2.91c)$$

**Convolution in Time** The DTFT pair corresponding to convolution in time is

$$(h * x)_n \quad \stackrel{\text{DTFT}}{\longleftrightarrow} \quad H(e^{j\omega}) X(e^{j\omega}). \quad (2.92)$$

First, a direct algebraic proof: The spectrum  $Y(e^{j\omega})$  of the output sequence  $y = h * x$  can be written as

$$\begin{aligned} Y(e^{j\omega}) &\stackrel{(a)}{=} \sum_{n \in \mathbb{Z}} y_n e^{-j\omega n} \stackrel{(b)}{=} \sum_{n \in \mathbb{Z}} \left( \sum_{k \in \mathbb{Z}} x_k h_{n-k} \right) e^{-j\omega n} \\ &= \sum_{n \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} x_k e^{-j\omega k} h_{n-k} e^{-j\omega(n-k)} \\ &\stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} x_k e^{-j\omega k} \sum_{n \in \mathbb{Z}} h_{n-k} e^{-j\omega(n-k)} \stackrel{(d)}{=} X(e^{j\omega}) H(e^{j\omega}), \end{aligned} \quad (2.93)$$

DTFT properties	Time domain	DTFT domain
<b>Basic properties</b>		
Linearity	$\alpha x_n + \beta y_n$	$\alpha X(e^{j\omega}) + \beta Y(e^{j\omega})$
Shift in time	$x_{n-n_0}$	$e^{-j\omega n_0} X(e^{j\omega})$
Shift in frequency	$e^{j\omega_0 n} x_n$	$X(e^{j(\omega-\omega_0)})$
Scaling in time		
Downsampling	$x_{Nn}$	$\frac{1}{N} \sum_{k=0}^{N-1} X(e^{j(\omega-2\pi k)/N})$
Upsampling	$x_{n/N}, n = \ell N; 0, \text{ otherwise}$	$X(e^{jN\omega})$
Time reversal	$x_{-n}$	$X(e^{-j\omega})$
Differentiation in freq.	$(-jn)^k x_n$	$\frac{\partial^k X(e^{j\omega})}{\partial \omega^k}$
Moments	$m_k = \sum_{n \in \mathbb{Z}} n^k x_n = (-j)^k \frac{\partial X(e^{j\omega})}{\partial \omega} \Big _{\omega=0}$	
Convolution in time	$(h * x)_n$	$H(e^{j\omega}) X(e^{j\omega})$
Convolution in frequency	$h_n x_n$	$\frac{1}{2\pi} (H \circledast X)(e^{j\omega})$
Deterministic autocorrelation	$a_n = \sum_{k \in \mathbb{Z}} x_k x_{k-n}^*$	$A(e^{j\omega}) =  X(e^{j\omega}) ^2$
Deterministic crosscorrelation	$c_n = \sum_{k \in \mathbb{Z}} x_k y_{k-n}^*$	$C(e^{j\omega}) = X(e^{j\omega}) Y^*(e^{j\omega})$
Parseval's equality	$\ x\ ^2 = \sum_{n \in \mathbb{Z}}  x_n ^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi}  X(e^{j\omega}) ^2 d\omega = \frac{1}{2\pi} \ X\ ^2$	
<b>Symmetries</b>		
Conjugate	$x_n^*$	$X^*(e^{-j\omega})$
Conjugate, time reversed	$x_{-n}^*$	$X^*(e^{j\omega})$
Real part	$\Re(x_n)$	$(X(e^{j\omega}) + X^*(e^{-j\omega}))/2$
Imaginary part	$\Im(x_n)$	$(X(e^{j\omega}) - X^*(e^{-j\omega}))/2j$
Conjugate-symmetric part	$(x_n + x_{-n}^*)/2$	$\Re(X(e^{j\omega}))$
Conjugate-antisymmetric part	$(x_n - x_{-n}^*)/2j$	$\Im(X(e^{j\omega}))$
<b>Symmetries for real <math>x</math></b>		
$X$ conjugate symmetric		$X(e^{j\omega}) = X^*(e^{-j\omega})$
Real part of $X$ even		$\Re(X(e^{j\omega})) = \Re(X(e^{-j\omega}))$
Imaginary part of $X$ odd		$\Im(X(e^{j\omega})) = -\Im(X(e^{-j\omega}))$
Magnitude of $X$ even		$ X(e^{j\omega})  =  X(e^{-j\omega}) $
Phase of $X$ odd		$\arg X(e^{j\omega}) = -\arg X(e^{-j\omega})$
<b>Common transform pairs</b>		
Kronecker delta sequence	$\delta_n$	1
Shift by $k$	$\delta_{n-k}$	$e^{-j\omega k}$
Constant	1	$2\pi \delta(\omega)$
Exponential sequence	$\alpha^n u_n$	$1/(1 - \alpha e^{-j\omega}) \quad  \alpha  < 1$
Differentiation	$n \alpha^n u_n$	$\alpha e^{-j\omega} / (1 - \alpha e^{-j\omega})^2 \quad  \alpha  < 1$
Ideal lowpass filter	$\sqrt{\frac{\omega_0}{2\pi}} \text{sinc}(\omega_0 n/2)$	$\begin{cases} \sqrt{2\pi/\omega_0}, & \text{for }  \omega  \leq \omega_0/2; \\ 0, & \text{otherwise.} \end{cases}$
Box sequence	$\begin{cases} 1/\sqrt{n_0}, & \text{for }  n  \leq (n_0 - 1)/2; \\ 0, & \text{otherwise,} \end{cases}$	$\sqrt{n_0} \frac{\text{sinc}(\omega n_0/2)}{\text{sinc}(\omega/2)}$

Table 2.4: Properties of the DTFT.

where (a) follows from the definition of the DTFT; (b) from the definition of convolution; (c) from interchanging the order of summation, an allowed operation since absolute summability is implied by  $h * x$  being well defined; and (d) from the definition of the DTFT.

This key result is a direct consequence of the eigensequence property of complex exponential sequences  $v$  from (2.75): when  $x$  is written as a combination of spectral components, each spectral component is simply scaled by the corresponding eigenvalue of the convolution operator; thus, using the DTFT has diagonalized the convolution operator.

**Convolution in Frequency** The DTFT pair corresponding to convolution in frequency is

$$h_n x_n \xleftrightarrow{\text{DTFT}} \frac{1}{2\pi} (H \circledast X)(e^{j\omega}), \quad (2.94)$$

where we have introduced the circular convolution between  $2\pi$ -periodic functions

$$(H \circledast X)(e^{j\omega}) = \int_{-\pi}^{\pi} X(e^{j\theta}) H(e^{j(\omega-\theta)}) d\theta. \quad (2.95)$$

Convolution in frequency is often referred to as *modulation in time*, and it is dual to convolution in time (see Exercise 2.6).

**Deterministic Autocorrelation** The DTFT pair corresponding to the deterministic autocorrelation of a sequence  $x$  is

$$a_n = \sum_{k \in \mathbb{Z}} x_k x_{k-n}^* \xleftrightarrow{\text{DTFT}} A(e^{j\omega}) = |X(e^{j\omega})|^2 \quad (2.96)$$

and satisfies

$$A(e^{j\omega}) = A^*(e^{j\omega}), \quad (2.97a)$$

$$A(e^{j\omega}) \geq 0. \quad (2.97b)$$

Thus,  $A(e^{j\omega})$  is not only real, (2.97a), but positive semidefinite as well, (2.97b). To verify (2.96), express the deterministic autocorrelation as a convolution of  $x$  and its time-reversed version as in (2.61d),  $x_n * x_{-n}^*$ . We know from Table 2.4 that the DTFT of  $x_{-n}^*$  is  $X^*(e^{j\omega})$ . Then, using the convolution property (2.92), we obtain (2.96).

For a real  $x$ ,

$$A(e^{j\omega}) = |X(e^{j\omega})|^2 = A(e^{-j\omega}), \quad (2.97c)$$

since  $X(e^{-j\omega}) = X^*(e^{j\omega})$ . The quantity  $A(e^{j\omega})$  is often called *energy spectral density* (the deterministic counterpart of the *power spectral density* for WSS sequences<sup>47</sup> in (2.232)). The *energy* is the integral of the energy spectral density over the frequency range,

$$E = \frac{1}{2\pi} \int_{-\pi}^{\pi} A(e^{j\omega}) d\omega = \frac{1}{2\pi} \int_{-\pi}^{\pi} |X(e^{j\omega})|^2 d\omega = \sum_{n \in \mathbb{Z}} |x_n|^2 = a_0. \quad (2.98)$$

<sup>47</sup>WSS stands for *wide-sense stationary*, defined in (2.224).

Thus, the energy spectral density measures the distribution of energy over the frequency range. Mimicking the relationship between the energy spectral density for deterministic sequences and the power spectral density for WSS sequences, (2.98) is the deterministic counterpart of the *power* for WSS sequences (2.233).

**Deterministic Crosscorrelation** The DTFT pair corresponding to the deterministic crosscorrelation of sequences  $x$  and  $y$  is

$$c_n = \sum_{k \in \mathbb{Z}} x_k y_{k-n}^* \xleftrightarrow{\text{DTFT}} C_{x,y}(e^{j\omega}) = X(e^{j\omega}) Y^*(e^{j\omega}), \quad (2.99)$$

and satisfies

$$C_{x,y}(e^{j\omega}) = C_{y,x}^*(e^{j\omega}). \quad (2.100a)$$

For  $x, y$  real,

$$C_{x,y}(e^{j\omega}) = X(e^{j\omega}) Y(e^{-j\omega}) = C_{y,x}(e^{-j\omega}). \quad (2.100b)$$

Further properties of deterministic autocorrelation and crosscorrelation sequences and their transforms are explored in Exercise 2.5.

**Deterministic Autocorrelation of Vector Sequences** The DTFT pair corresponding to the deterministic autocorrelation of a vector sequence  $x$  is

$$A_n \xleftrightarrow{\text{DTFT}} A(e^{j\omega}) = \begin{bmatrix} A_0(e^{j\omega}) & C_{0,1}(e^{j\omega}) & \dots & C_{0,N-1}(e^{j\omega}) \\ C_{1,0}(e^{j\omega}) & A_1(e^{j\omega}) & \dots & C_{1,N-1}(e^{j\omega}) \\ \vdots & \vdots & \ddots & \vdots \\ C_{N-1,0}(e^{j\omega}) & C_{N-1,1}(e^{j\omega}) & \dots & A_{N-1}(e^{j\omega}) \end{bmatrix}, \quad (2.101)$$

where  $A_n$  is given in (2.22). Because of (2.97a) and (2.100a), this energy spectral density matrix is Hermitian, that is,

$$A(e^{j\omega}) = \begin{bmatrix} A_0(e^{j\omega}) & C_{0,1}(e^{j\omega}) & \dots & C_{0,N-1}(e^{j\omega}) \\ C_{0,1}^*(e^{j\omega}) & A_1(e^{j\omega}) & \dots & C_{1,N-1}(e^{j\omega}) \\ \vdots & \vdots & \ddots & \vdots \\ C_{0,N-1}^*(e^{j\omega}) & C_{1,N-1}^*(e^{j\omega}) & \dots & A_{N-1}(e^{j\omega}) \end{bmatrix} = A^*(e^{j\omega}). \quad (2.102a)$$

For a real  $x$ ,

$$A(e^{j\omega}) = A^T(e^{-j\omega}). \quad (2.102b)$$

**Parseval's Equality** As noted earlier, from the form of (2.78a), the DTFT is a linear operator from the space of sequences to the space of  $2\pi$ -periodic functions. Let us denote this through  $X = Fx$ . We have  $F : \ell^2(\mathbb{Z}) \rightarrow \mathcal{L}^2([-\pi, \pi])$  because

$x \in \ell^2(\mathbb{Z})$  implies that  $X(e^{j\omega})$  has finite  $\mathcal{L}^2([-\pi, \pi])$  norm. Specifically,

$$\begin{aligned}
 \|X\|^2 &\stackrel{(a)}{=} \int_{-\pi}^{\pi} |X(e^{j\omega})|^2 d\omega = \int_{-\pi}^{\pi} X(e^{j\omega}) X^*(e^{j\omega}) d\omega \\
 &\stackrel{(b)}{=} \int_{-\pi}^{\pi} \left( \sum_{n \in \mathbb{Z}} x_n e^{j\omega n} \right) \left( \sum_{k \in \mathbb{Z}} x_k e^{j\omega k} \right)^* d\omega \\
 &\stackrel{(c)}{=} \sum_{n \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} \int_{-\pi}^{\pi} x_n x_k^* e^{j\omega(n-k)} d\omega = \sum_{n \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} x_n x_k^* \int_{-\pi}^{\pi} e^{j\omega(n-k)} d\omega \\
 &\stackrel{(d)}{=} \sum_{n \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} x_n x_k^* 2\pi \delta_{n-k} \stackrel{(e)}{=} 2\pi \sum_{n \in \mathbb{Z}} x_n x_n^* \\
 &= 2\pi \sum_{n \in \mathbb{Z}} |x_n|^2 \stackrel{(f)}{=} 2\pi \|x\|^2,
 \end{aligned} \tag{2.103}$$

where (a) follows from the definition of the  $\mathcal{L}^2([-\pi, \pi])$  norm; (b) from the definition of the DTFT; (c) from an interchange that is allowed because  $x \in \ell^2(\mathbb{Z})$  implies absolute convergence of the sums in the integrand; (d) from (2.80b); (e) from the definition of the Kronecker delta sequence, (1.9); and (f) from the definition of the  $\ell^2(\mathbb{Z})$  norm.

If it were not for the  $2\pi$  factor, the equality (2.103) would be like the equality (1.51) for a unitary operator; (2.103) is the version of *Parseval's equality* for the DTFT. Parseval's equality<sup>48</sup> is often termed the *energy conservation property*, as the energy (2.98) is the integral of the energy spectral density over the frequency range.

A computation similar to (2.103) shows that  $F/\sqrt{2\pi}$  is a unitary operator (see (1.50)):

$$\left\langle \frac{1}{\sqrt{2\pi}} Fx, \frac{1}{\sqrt{2\pi}} Fy \right\rangle = \langle x, y \rangle \quad \text{for every } x \text{ and } y \text{ in } \ell^2(\mathbb{Z}),$$

or, equivalently,

$$\langle x, y \rangle = \frac{1}{2\pi} \langle X, Y \rangle \quad \text{for every } x \text{ and } y \text{ in } \ell^2(\mathbb{Z}), \tag{2.104}$$

where  $X$  and  $Y$  are the DTFTs of  $x$  and  $y$ . This is the version of the *generalized Parseval's equality* for the DTFT, and its proof is left as Exercise 2.5.

**Adjoint** The adjoint of the DTFT,  $F^* : \mathcal{L}^2([-\pi, \pi]) \rightarrow \ell^2(\mathbb{Z})$ , is determined uniquely by

$$\langle Fx, y \rangle = \langle x, F^*y \rangle \quad \text{for every } x \in \ell^2(\mathbb{Z}) \text{ and } y \text{ in } \mathcal{L}^2([-\pi, \pi]).$$

<sup>48</sup>Recall that what we call Parseval's equality in this book is sometimes called Plancherel's equality; what we call generalized Parseval's equality is sometimes called Parseval's theorem.

Since we have already concluded that  $F/\sqrt{2\pi}$  is a unitary operator, by Theorem 1.23,

$$\left(\frac{1}{\sqrt{2\pi}}F\right)^* = \left(\frac{1}{\sqrt{2\pi}}F\right)^{-1} = \sqrt{2\pi}F^{-1}.$$

Thus,  $F^* = 2\pi F^{-1}$ , with  $F^{-1}$  given by (2.78b).

#### 2.4.4 Frequency Response of Filters

The DTFT of a sequence is called its spectrum. The DTFT of a filter (impulse response of an LSI system)  $h$  is also called the *frequency response*:

$$H(e^{j\omega}) = \sum_{n \in \mathbb{Z}} h_n e^{-j\omega n}, \quad \omega \in \mathbb{R}. \quad (2.105a)$$

The inverse DTFT of the frequency response recovers the impulse response:

$$h_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(e^{j\omega}) e^{j\omega n} d\omega, \quad n \in \mathbb{Z}. \quad (2.105b)$$

To understand the frequency response of a filter, we often write the magnitude and phase separately:

$$H(e^{j\omega}) = |H(e^{j\omega})| e^{j \arg(H(e^{j\omega}))},$$

where the *magnitude response*  $|H(e^{j\omega})|$  is a  $2\pi$ -periodic real, nonnegative function, and the *phase response*  $\arg(H(e^{j\omega}))$  is a  $2\pi$ -periodic real function between  $-\pi$  and  $\pi$ .<sup>49</sup> A filter is said to have *zero phase* when its frequency response is real; this is equivalent to the phase response taking only values that are integer multiples of  $\pi$ . A filter is said to have *generalized linear phase* when its frequency response can be written in the form

$$H(e^{j\omega}) = r(\omega) e^{j(\alpha\omega + \beta)}, \quad (2.106)$$

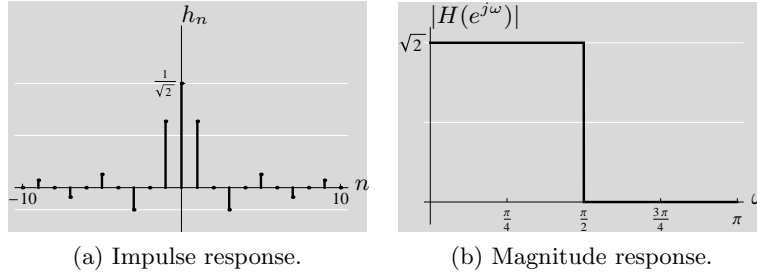
where  $r(\omega)$  is real and  $\alpha$  and  $\beta$  are constants; this corresponds to a phase response that is affine in  $\omega$  (straight lines with slope  $\alpha$ ) except where there are jumps by  $2\pi$ . When furthermore  $\beta = 0$ , the filter is said to have *linear phase*. A filter is called *bandlimited* when its frequency response is finitely supported. Solved Exercise 2.3 explores filters as projections through their frequency response.

**Ideal Filters** The frequency response of a filter is typically used to design a filter with specific properties, where we want to let certain frequencies pass—the *passband*, while blocking others—the *stopband*. An *ideal filter* is a filter whose magnitude response takes a single nonzero value in its passband. For example, an ideal lowpass filter passes frequencies below some cut-off frequency  $\omega_0/2$  and blocks the others; its passband is thus the interval  $[-\omega_0/2, \omega_0/2]$ . All ideal filters are bandlimited. Figure 2.9(b) gives an example for  $\omega_0 = \pi$ .

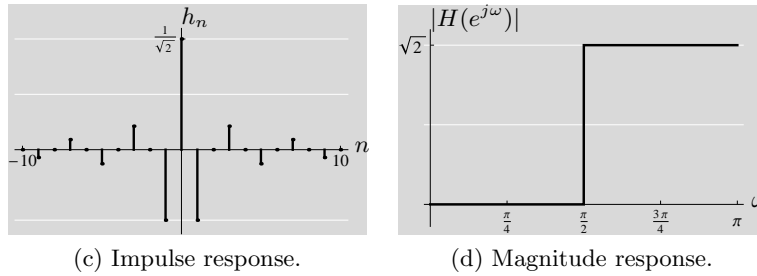
<sup>49</sup>The argument of the complex number  $H(e^{j\omega})$  can be equally well defined to be on  $[0, 2\pi)$ .



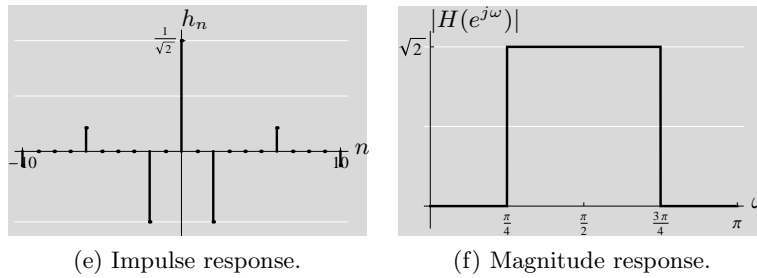
## Ideal lowpass filter



## Ideal highpass filter



## Ideal bandpass filter

**Figure 2.9:** Impulse and magnitude responses of ideal filters.

To find the impulse response of such an ideal filter, we start with the desired magnitude response:

$$H(e^{j\omega}) = \begin{cases} \sqrt{2\pi/\omega_0}, & \text{for } |\omega| \leq \omega_0/2; \\ 0, & \text{otherwise.} \end{cases} \quad (2.107a)$$

This is a zero-phase filter and a box function in frequency.<sup>50</sup> Applying the inverse

<sup>50</sup>Table 3.6 in Chapter 3 summarizes box and sinc functions in time and frequency.

DTFT, we obtain the impulse response as

$$h_n = \frac{1}{\sqrt{2\pi\omega_0}} \int_{-\omega_0/2}^{\omega_0/2} e^{j\omega n} d\omega = \sqrt{\frac{\omega_0}{2\pi}} \operatorname{sinc}(\omega_0 n/2) \quad (2.107b)$$

by elementary integrations, with the  $n = 0$  and  $n \neq 0$  cases separated. This impulse response is of unit norm. A case of interest is the  $N$ th band filter  $\omega_0 = 2\pi/N$ , and in particular, a halfband filter when  $\omega_0 = \pi$ , which passes through half of the spectrum, from  $-\pi/2$  to  $\pi/2$ , with magnitude response as in Figure 2.9(b), and impulse response

$$h_n = \frac{1}{\sqrt{2}} \operatorname{sinc}(\pi n/2) = \frac{1}{\sqrt{2}} \frac{\sin(\pi n/2)}{\pi n/2} \quad (2.108)$$

as in Figure 2.9(a). These ideal filters are summarized in Table 2.5. Their impulse responses decay slowly as  $O(1/n)$  and are thus not absolutely summable. This lack of absolute summability of the impulse response  $h$  is unavoidable when the desired frequency response  $H$  is discontinuous; see Section 2.4.2.

Ideal filters	Time domain	DTFT domain
Ideal lowpass filter	$\sqrt{\frac{\omega_0}{2\pi}} \operatorname{sinc}(\omega_0 n/2)$	$\begin{cases} \sqrt{2\pi/\omega_0}, &  \omega  \leq \omega_0/2; \\ 0, & \text{otherwise.} \end{cases}$
Ideal $N$ th-band filter	$(1/\sqrt{N}) \operatorname{sinc}(\pi n/N)$	$\begin{cases} \sqrt{N}, &  \omega  \leq \pi/N; \\ 0, & \text{otherwise.} \end{cases}$
Ideal halfband lowpass filter	$(1/\sqrt{2}) \operatorname{sinc}(\pi n/2)$	$\begin{cases} \sqrt{2}, &  \omega  \leq \pi/2; \\ 0, & \text{otherwise.} \end{cases}$

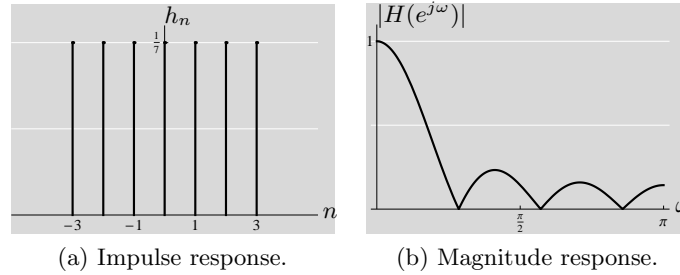
**Table 2.5:** Ideal filters with unit-norm impulse responses.

**FIR Filters** Ideal filters are not realizable; thus, we now explore a few examples of filters with realizable frequency responses. We start with an FIR filter we have already seen in Example 2.2.

**EXAMPLE 2.15 (MOVING AVERAGE FILTER, EXAMPLE 2.2 CONT'D)** The impulse response of the moving average filter in (2.5) is (we assumed  $N$  odd):

$$h_n = \begin{cases} 1/N, & \text{for } |n| \leq (N-1)/2; \\ 0, & \text{otherwise,} \end{cases} \quad (2.109a)$$

which is the same, within scaling, as the box sequence from (2.13a). Its frequency



**Figure 2.10:** Moving average filter (2.5) with  $N = 7$ .

response is

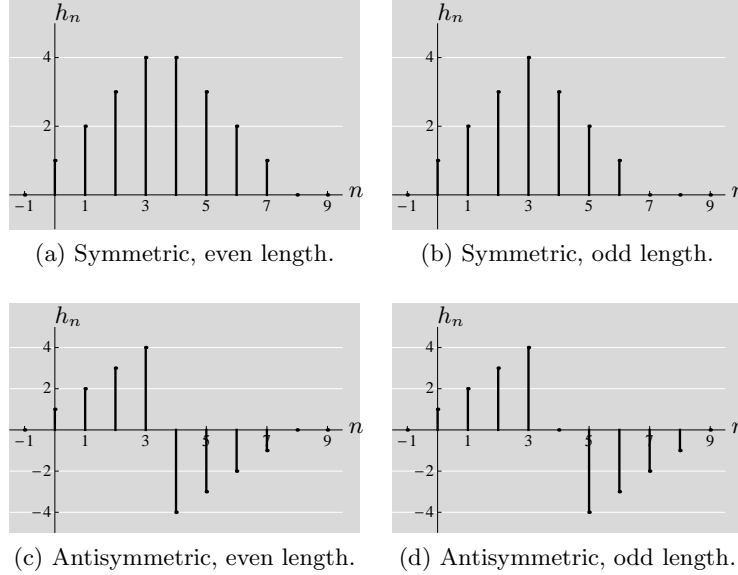
$$\begin{aligned}
 H(e^{j\omega}) &= \frac{1}{N} \sum_{n=-(N-1)/2}^{(N-1)/2} e^{-j\omega n} \stackrel{(a)}{=} \frac{1}{N} e^{j\omega(N-1)/2} \sum_{k=0}^{N-1} e^{-j\omega k} \\
 &\stackrel{(b)}{=} \frac{1}{N} e^{j\omega(N-1)/2} \frac{1 - e^{-j\omega N}}{1 - e^{-j\omega}} = \frac{1}{N} \frac{e^{j\omega N/2} - e^{-j\omega N/2}}{e^{j\omega/2} - e^{-j\omega/2}} \\
 &= \frac{1}{N} \frac{\sin(\omega N/2)}{\sin(\omega/2)}, \tag{2.109b}
 \end{aligned}$$

where (a) follows from change of variable  $k = n + (N - 1)/2$ ; and (b) from (P1.65-1), the formula for a finite geometric series. Figure 2.10 shows the impulse response and magnitude response of this filter for  $N = 7$ .

**Linear-Phase Filters** Real-valued FIR filters have linear phase when they are symmetric or antisymmetric. Consider causal filters with length  $L$ , so the support is  $\{0, 1, \dots, L - 1\}$ . These filters then satisfy

$$\begin{array}{ll}
 \text{symmetric} & \text{antisymmetric} \\
 h_n = h_{L-1-n} & h_n = -h_{L-1-n}
 \end{array} \tag{2.110}$$

These symmetries are illustrated in Figure 2.11 for  $L$  even and odd. Let us now show that an even-length, symmetric filter as in part (a) of the figure indeed leads to linear phase; other cases follow similarly. We compute the frequency response of

**Figure 2.11:** Filters with symmetries.

$h_n$ ,

$$\begin{aligned}
 H(e^{j\omega}) &= \sum_{n=0}^{L-1} h_n e^{-j\omega n} \stackrel{(a)}{=} \sum_{n=0}^{L/2-1} h_n \left( e^{-j\omega n} + e^{-j\omega(L-1-n)} \right) \\
 &= \sum_{n=0}^{L/2-1} h_n e^{-j\omega(L-1)/2} \left( e^{j\omega(n-(L-1)/2)} + e^{-j\omega(n-(L-1)/2)} \right) \\
 &\stackrel{(b)}{=} 2 \sum_{n=0}^{L/2-1} h_n \cos \left( \omega \left( n - \frac{L-1}{2} \right) \right) e^{-j\omega((L-1)/2)} \\
 &= r(\omega) e^{j\alpha\omega}, \tag{2.111a}
 \end{aligned}$$

with

$$r(\omega) = 2 \sum_{n=0}^{L/2-1} h_n \cos \left( \omega \left( n - \frac{L-1}{2} \right) \right) \quad \text{and} \quad \alpha = -\frac{L-1}{2}. \tag{2.111b}$$

This frequency response fits the form of (2.106), so the filter indeed has linear phase. In the above, (a) follows from gathering factors with the same  $h_n$  because of symmetry in (2.110); and (b) from using (2.275).

**Allpass Filters** Another important class is filters with unit magnitude response, that is,

$$|H(e^{j\omega})| = 1. \tag{2.112}$$

## 2.4. Discrete-Time Fourier Transform

221

Since all frequencies go through without change of magnitude, a filter satisfying (2.112) is called an *allpass filter*. Allpass filters have some interesting properties:

- (i) *Energy conservation*: The allpass property corresponds to energy conservation, since, using Parseval's equality (2.103), we have

$$\begin{aligned}\|y\|^2 &= \frac{1}{2\pi} \|Y(e^{j\omega})\|^2 = \frac{1}{2\pi} \|H(e^{j\omega})X(e^{j\omega})\|^2 \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} |H(e^{j\omega})X(e^{j\omega})|^2 d\omega = \frac{1}{2\pi} \int_{-\pi}^{\pi} |X(e^{j\omega})|^2 d\omega = \|x\|^2.\end{aligned}$$

- (ii) *Orthonormal set*: The allpass property implies that all the shifts of  $h$ ,  $\{\varphi_{k,n} = h_{n-k}\}_{k \in \mathbb{Z}}$ , form an orthonormal set:

$$\begin{aligned}\langle h_n, h_{n-k} \rangle_n &= \sum_{n \in \mathbb{Z}} h_n h_{n-k}^* \stackrel{(a)}{=} \frac{1}{2\pi} \int_{-\pi}^{\pi} H(e^{j\omega}) (e^{-j\omega k} H(e^{j\omega}))^* d\omega \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j\omega k} H(e^{j\omega}) H^*(e^{j\omega}) d\omega \stackrel{(b)}{=} \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j\omega k} \underbrace{|H(e^{j\omega})|^2}_1 d\omega \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j\omega k} d\omega \stackrel{(c)}{=} \frac{\sin k\pi}{k\pi} \stackrel{(d)}{=} \delta_k,\end{aligned}\tag{2.113}$$

where (a) follows from the generalized Parseval's equality (2.104) and the shift in time property (2.85); (b) from our assumption; (c) from (2.107b) with  $\omega_0 = 2\pi$  and scaling of the filter's magnitude response; and (d) from (2.8c). We summarize this property as

$$\langle h_n, h_{n-k} \rangle_n = \delta_k \quad \xleftrightarrow{\text{DTFT}} \quad |H(e^{j\omega})| = 1.\tag{2.114}$$

- (iii) *Orthonormal basis*: The allpass property implies that  $\{\varphi_{k,n} = h_{n-k}\}_{k \in \mathbb{Z}}$  is an orthonormal basis for  $\ell^2(\mathbb{Z})$ . Having already shown in (2.113) that the set is orthonormal, we check if we can write any  $x_n$  as

$$x_n = \sum_{k \in \mathbb{Z}} \beta_k \varphi_{k,n} = \sum_{k \in \mathbb{Z}} \beta_k h_{n-k},$$

with  $\beta_k = \langle x_n, h_{n-k} \rangle_n$ . It is sufficient to verify that  $\|\beta\| = \|x\|$ . Write

$$\beta_k = \sum_{n \in \mathbb{Z}} x_n h_{n-k}^* = \sum_{n \in \mathbb{Z}} x_n h_{-(k-n)}^* = x_n * h_{k-n}^*,$$

and thus,

$$\|\beta\|^2 \stackrel{(a)}{=} \frac{1}{2\pi} \|X(e^{j\omega})H^*(e^{j\omega})\|^2 \stackrel{(b)}{=} \frac{1}{2\pi} \|X(e^{j\omega})\|^2 \stackrel{(c)}{=} \|x\|^2,$$

where (a) follows from the convolution property (2.92), Parseval's equality (2.103) and (2.89); (b) from  $H(e^{j\omega})$  having unit magnitude for all  $\omega$ ; and (c)

from Parseval's equality again. Figure 2.12 shows the phase of  $H(e^{j\omega})$  given in (2.115).

This discussion contains a piece of good news—there exist shift-invariant orthonormal bases for  $\ell^2(\mathbb{Z})$ , as well as a piece of bad news—these bases have no frequency selectivity (they are allpass sequences). This is one of the main reasons to search for more general orthonormal bases for  $\ell^2(\mathbb{Z})$ , as we do in Part II of the book.

**EXAMPLE 2.16 (ALLPASS FILTERS)** Consider the simple shift-by- $k$  filter given in (2.38a) with the impulse response  $h_n = \delta_{n-k}$ . By evaluating (2.105a), the frequency response is  $H(e^{j\omega}) = e^{-j\omega k}$ . Thus,  $h$  is an allpass filter:

$$|H(e^{j\omega})| = 1, \quad \arg(H(e^{j\omega})) = -\omega k \bmod 2\pi.$$

This filter has linear phase with a slope  $-k$  given by the delay.

We now look at a more sophisticated allpass filter. It provides an example where also see that while key properties that are not plainly visible in the time domain become obvious in the frequency domain. The filter is:

$$g_n = \alpha^n u_n, \quad g = \left[ \dots \quad 0 \quad \boxed{1} \quad \alpha \quad \alpha^2 \quad \alpha^3 \quad \dots \right]^T,$$

with  $\alpha \in \mathbb{C}$  and  $|\alpha| < 1$ , and  $u_n$  is the Heaviside sequence from (2.10). Suppose  $h$  satisfies

$$h_n = -\alpha^* g_n + g_{n-1}, \quad n \in \mathbb{Z}.$$

We now show  $h$  is an allpass filter, so filtering a sequence  $x$  with  $h$  will not change its magnitude; moreover,  $h$  is of norm 1 and orthogonal to all its shifts as in (2.114). To start, find the frequency response of  $g_n$ ,

$$G(e^{j\omega}) = \sum_{n \in \mathbb{Z}} \alpha^n e^{-j\omega n} \stackrel{(a)}{=} \frac{1}{1 - \alpha e^{-j\omega}},$$

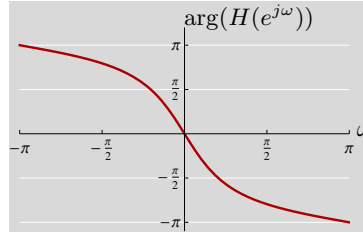
where (a) follows from the formula for the infinite geometric series (P1.65-3). Then,

$$H(e^{j\omega}) = -\alpha^* G(e^{j\omega}) + e^{-j\omega} G(e^{j\omega}) = \frac{e^{-j\omega} - \alpha^*}{1 - \alpha e^{-j\omega}}. \quad (2.115)$$

The magnitude squared of  $H(e^{j\omega})$  is

$$|H(e^{j\omega})|^2 = H(e^{j\omega}) H^*(e^{j\omega}) = \frac{(e^{-j\omega} - \alpha^*)(e^{j\omega} - \alpha)}{(1 - \alpha e^{-j\omega})(1 - \alpha^* e^{j\omega})} = 1,$$

and thus  $|H(e^{j\omega})| = 1$  for all  $\omega$ . The phase response is shown in Figure 2.12.



**Figure 2.12:** Phase of a first-order allpass filter as in (2.115) with  $\alpha = 1/2$ .

## 2.5 $z$ -Transform

While the DTFT has many nice properties, its use is limited by the convergence issues discussed in Section 2.4.2. The  $z$ -transform introduces a set of rescalings of sequences so that almost any sequence has a rescaling such that the DTFT converges. This makes the  $z$ -transform much more widely applicable than the DTFT.

Take the Heaviside sequence from (2.10), which is neither in  $\ell^1(\mathbb{Z})$  nor  $\ell^2(\mathbb{Z})$  (nor any other  $\ell^p(\mathbb{Z})$  space except  $\ell^\infty(\mathbb{Z})$ ), and thus has no DTFT. If we were to multiply it by a geometric sequence  $r^n$ , with  $r \in [0, 1)$ , yielding  $x_n = r^n u_n$ , we could take the DTFT of  $x_n$ , as we now have an absolutely-summable sequence. By controlling the rescaling with  $r$ , we have a set of DTFTs indexed by  $r$ , and we can think of this as a new transform with arguments  $r$  and  $\omega$ . Combining  $r$  and  $\omega$  through  $z = re^{j\omega}$ , we obtain a transform with argument  $z \in \mathbb{C}$  where  $z$  need not have unit modulus.

Because of the close connection to the DTFT, we will have a convolution property as well as many other properties similar to those in Section 2.4.3, but now for more general sequences. Indeed, as we will see shortly, convolution of finite-length sequences becomes polynomial multiplication in the  $z$ -transform domain. This is the essential motivation behind extending the analysis that uses the unit-norm complex exponential sequences in (2.75) to more general complex exponential sequences  $v_n = z^n = (re^{j\omega})^n$ .

### 2.5.1 Definition of the $z$ -Transform

**Eigensequences of the Convolution Operator** The eigensequence property (2.76) extends from complex exponentials with unit modulus to those with any modulus. Consider the sequence

$$v_n = z^n = (re^{j\omega})^n, \quad n \in \mathbb{Z}, \quad (2.116)$$

where  $r \in [0, \infty)$  and  $\omega \in \mathbb{R}$ , so  $z$  is any complex number. Like a complex exponential sequence with unit modulus, this is also an eigensequence of the convolution

operator  $H$  associated with the LSI system with impulse response  $h$  since

$$\begin{aligned} (Hv)_n &= (h * v)_n = \sum_{k \in \mathbb{Z}} v_{z, n-k} h_k = \sum_{k \in \mathbb{Z}} z^{n-k} h_k \\ &= \underbrace{\sum_{k \in \mathbb{Z}} h_k z^{-k}}_{\lambda_z} \underbrace{z^n}_{v_n}. \end{aligned} \quad (2.117)$$

This shows that applying the convolution operator  $H$  to the sequence  $v$  gives a scalar multiple of  $v$ ;  $v$  is an eigensequence of  $H$  with corresponding eigenvalue  $\lambda_z$ . We call that eigenvalue  $H(z)$ ; it is defined formally in (2.150). We can thus rewrite (2.117) as

$$H z^n = h * z^n = H(z) z^n. \quad (2.118)$$

The key distinction from (2.76) is that the set of impulse responses  $h$  for which the sum (2.117) converges now depends on  $|z|$ .

**$z$ -Transform** The  $z$ -transform is defined similarly to the DTFT in Definition 2.11:

**DEFINITION 2.12 ( $z$ -TRANSFORM)** The  $z$ -transform of a sequence  $x$  is

$$X(z) = \sum_{n \in \mathbb{Z}} x_n z^{-n}, \quad z \in \mathbb{C}. \quad (2.119)$$

It exists when (2.119) converges absolutely for some values of  $z$ ; these values of  $z$  are called the *region of convergence (ROC)*,

$$\text{ROC} = \{z \mid |X(z)| < \infty\}. \quad (2.120)$$

When the  $z$ -transform exists, we denote the  $z$ -transform pair as

$$x_n \xleftrightarrow{\text{ZT}} X(z),$$

where the ROC is part of the specification of  $X(z)$ .

**Relation of the  $z$ -Transform to the DTFT** Given a sequence  $x$  and its  $z$ -transform  $X(z)$  with an ROC that includes the unit circle  $|z| = 1$ , the  $z$ -transform evaluated on the unit circle is equal to the DTFT of the same sequence:

$$X(z)|_{z=e^{j\omega}} = X(e^{j\omega}). \quad (2.121)$$

Conversely, suppose  $y_n = r^{-n} x_n$  has DTFT  $Y(e^{j\omega})$ . Then

$$Y(e^{j\omega}) = \sum_{n \in \mathbb{Z}} r^{-n} x_n e^{-j\omega n} = \sum_{n \in \mathbb{Z}} x_n (re^{j\omega})^{-n} = X(re^{j\omega}),$$



so

$$X(z)|_{z=re^{j\omega}} = Y(e^{j\omega}) \quad (2.122)$$

and the circle  $|z| = r$  is in the ROC of  $X(z)$ .

### 2.5.2 Existence and Convergence of the $z$ -Transform

**Convergence** For the  $z$ -transform to exist and have  $z = re^{j\omega}$  in its ROC, (2.119) must converge absolutely. Since

$$\sum_{n \in \mathbb{Z}} |x_n z^{-n}| = \sum_{n \in \mathbb{Z}} |x_n r^{-n}| |e^{-j\omega n}| = \sum_{n \in \mathbb{Z}} |x_n r^{-n}|,$$

absolute summability of  $x_n r^{-n}$  is necessary and sufficient for the circle  $|z| = r$  to be in the ROC of  $X(z)$ . Thus, the ROC is a ring of the form (see also Table 2.6)

$$\text{ROC} = \{z \mid 0 \leq r_1 < |z| < r_2 \leq \infty\}. \quad (2.123)$$

By convention, the ROC concept is extended to  $|z| = \infty$  by including  $|z| = \infty$  in the ROC when  $x_n = 0$  for all  $n < 0$  and excluding it otherwise. Similarly,  $z = 0$  is in the ROC when  $x_n = 0$  for all  $n > 0$  and not in the ROC otherwise. Exercise 2.9 explores a number of properties of the ROC.

EXAMPLE 2.17 To develop intuition, we look at a few examples:

(i) *Shift-by- $n_0$  sequence*

$$x_n = \delta_{n-n_0} \xleftrightarrow{\text{ZT}} X(z) = z^{-n_0} \quad \text{ROC} = \begin{cases} |z| > 0, & \text{if } n_0 > 0; \\ \text{all } z, & \text{if } n_0 = 0; \\ |z| < \infty, & \text{if } n_0 < 0. \end{cases}$$

The shift-by-one maps to  $z^{-1}$ , which is why  $z^{-1}$  is often called a delay operator. It also follows that

$$x_{n-n_0} \xleftrightarrow{\text{ZT}} z^{-n_0} X(z) \quad \text{ROC} = \text{ROC}_x,$$

with the only possible changes to the ROC at 0 or  $\infty$ .

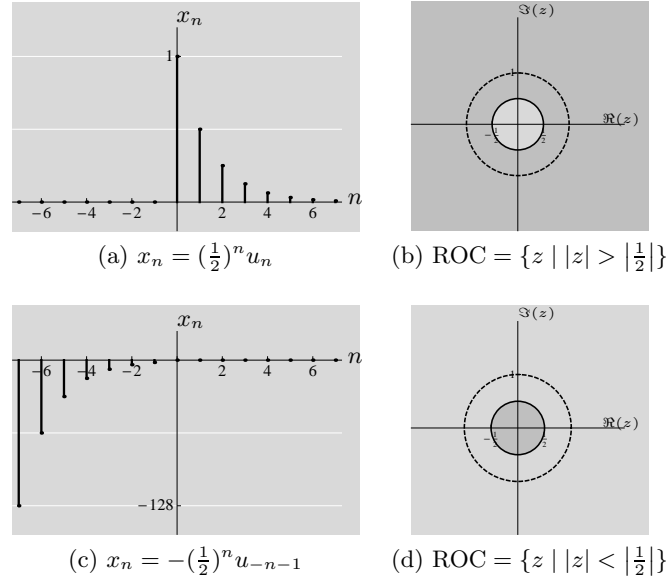
(ii) *Right-sided geometric sequence*

$$x_n = \alpha^n u_n \xleftrightarrow{\text{ZT}} X(z) = \frac{1}{1 - \alpha z^{-1}} \quad \text{ROC} = \{z \mid |z| > |\alpha|\}. \quad (2.124a)$$

Now,  $z = \alpha$  is a zero of the denominator of the complex function  $X(z)$ , and we see that the ROC is bounded from inside by a circle containing the  $z = \alpha$ . This is a general property, since the ROC cannot contain a singularity (a  $z$  such that  $X(z)$  does not exist).

(iii) *Left-sided geometric sequence*

$$x_n = -\alpha^n u_{-n-1} \xleftrightarrow{\text{ZT}} X(z) = \frac{1}{1 - \alpha z^{-1}} \quad \text{ROC} = \{z \mid |z| < |\alpha|\}. \quad (2.124b)$$



**Figure 2.13:** Illustration of Example 2.17. (a) The right-sided geometric series sequence and (b) the associated ROC of its  $z$ -transform. (c) The left-sided geometric series sequence and (d) the associated ROC of its  $z$ -transform. The unit circle is marked in both (b) and (d) for reference.

The expression for  $X(z)$  is exactly as in the previous case; the only difference is in the ROC. Had we been given only this  $X(z)$  without the associated ROC, we would not have been able to tell whether it originated from  $x$  in (2.124a) or (2.124b). This shows why the  $z$ -transform and its ROC form a pair that should not be broken.

A standard way of showing the ROC is a plot of the complex plane, as in Figure 2.13. Marking the unit circle establishes the scale of the plot, and the DTFT converges for all  $\omega$  when the unit circle is in the ROC.

**Rational  $z$ -Transforms** An important class of  $z$ -transforms are those that are rational functions, since transfer functions of most realizable systems (systems that can be built and used in practice) are rational. We will see in Section 2.5.4 that these are directly related to difference equations with a finite number of coefficients, as in (2.54). Such transfer functions are of the form

$$H(z) = \frac{B(z)}{A(z)}, \quad (2.125)$$

where  $A(z)$  and  $B(z)$  are polynomials in  $z^{-1}$ , of degree  $N$  and  $M$ , respectively. The degrees satisfy  $M \leq N$ , otherwise, polynomial division would lead to a sum

of a polynomial and a rational function satisfying this constraint. The zeros of the numerator  $B(z)$  and denominator  $A(z)$  are called the *zeros* and *poles* of the rational transfer function  $H(z)$ . Many properties of LSI systems depend on the zeros and poles and their multiplicities.

Consider a finite-length sequence  $h = [h_0 \ h_1 \ \dots \ h_M]^T$ . Then  $H(z) = \sum_{k=0}^M h_k z^{-k}$ , which has  $M$  poles at  $z = 0$  and  $M$  zeros at the roots  $\{z_k\}_{k=1}^M$  of the polynomial  $H(z)$ .<sup>51</sup> Therefore,  $H(z)$  can be written as

$$H(z) = h_0 \prod_{k=1}^M (1 - z_k z^{-1}), \quad |z| > 0, \quad (2.126)$$

where the form of the factorization shows explicitly both the roots as well as the multiplicative factor  $h_0$ .

The rational  $z$ -transform in (2.125) can thus also be written as

$$H(z) = \frac{b_0 \prod_{k=1}^M (1 - z_k z^{-1})}{a_0 \prod_{k=1}^N (1 - p_k z^{-1})}, \quad (2.127)$$

where  $\{z_k\}_{k=1}^M$  are zeros and  $\{p_k\}_{k=1}^N$  poles. The ROC cannot contain any poles and is thus, assuming a right-sided sequence, all  $z$  outside of the pole largest in magnitude. If  $M$  is smaller than  $N$ , then  $H(z)$  has  $N - M$  additional zeros at 0. This can be seen in our previous example (2.124a), which can be rewritten as  $1/(1 - \alpha z^{-1}) = z/(z - \alpha)$  and has thus a pole at  $z = \alpha$  and a zero at  $z = 0$ .

### Inversion

Given a  $z$ -transform and its ROC, how do we invert the  $z$ -transform? The general inversion formula for the  $z$ -transform involves contour integration, a standard topic of complex analysis. However, most  $z$ -transforms encountered in practice can be inverted using simpler methods which we now discuss; *Further Reading* gives pointers for a more detailed treatment of the inverse  $z$ -transform.

**Inversion by Inspection** This method is just a way of recognizing certain  $z$ -transform pairs. For example, from Table 2.6, we see that the  $z$ -transform

$$X(z) = \frac{1}{1 - (1/4)z^{-1}}$$

has the form of  $1/(1 - az^{-1})$ , with  $a = 1/4$ . From the table, we can then read the sequence that generated it as one of the following two:

$$\left(\frac{1}{4}\right)^n u_n, \quad \text{if ROC} = \{z \mid |z| > \frac{1}{4}\},$$

or

$$-\left(\frac{1}{4}\right)^n u_{-n-1}, \quad \text{if ROC} = \{z \mid |z| < \frac{1}{4}\}.$$

No other ROC is possible.

<sup>51</sup>The fundamental theorem of algebra (Theorem 2.19) states that a degree- $M$  polynomial has  $M$  complex roots.

**Inversion Using Partial Fraction Expansion** When the  $z$ -transform is given as a rational function, partial fraction expansion results in a sum of terms, each of which can be inverted by inspection. Here we consider cases in which the numerator and denominator are polynomials in  $z^{-1}$ , as in (2.127).

- (i)  $M < N$ , simple poles: If all the  $N$  poles are of first order, we can express  $X(z)$  as

$$X(z) = \sum_{k=1}^N \frac{A_k}{1 - p_k z^{-1}}, \quad A_k = (1 - p_k z^{-1})X(z)|_{z=p_k}. \quad (2.129a)$$

Each term has a simple inverse  $z$ -transform, which depends on the ROC of  $X(z)$ . The ROC takes one of the following forms:

$$\text{ROC} = \begin{cases} \{z \mid |z| < |p_1|\}; \\ \{z \mid |p_k| < |z| < |p_{k+1}|\} \text{ for some } k; \\ \{z \mid |z| > |p_N|\}, \end{cases}$$

where we have assumed  $|p_1| \leq |p_2| \leq \dots \leq |p_N|$  for simplicity. Each distinct ROC corresponds to a different sequence. Often the ROC is  $\{z \mid |z| > |p_N|\}$ , resulting in

$$x_n = \sum_{k=1}^N A_k (p_k)^n u_n. \quad (2.129b)$$

- (ii)  $M < N$ , poles with multiplicity: Suppose  $X(z)$  has pole  $p_i$  of order  $s > 1$ . Then, in general, the  $i$ th term in (2.129a) is replaced by  $s$  terms

$$\sum_{k=1}^s \frac{C_k}{(1 - p_i z^{-1})^k}.$$

The  $k = 1$  term is inverted as before, and the terms for  $k > 1$  are inverted using the differentiation rule from Table 2.6.

- (iii)  $M \geq N$ : Assume all poles are of first order; multiplicities can be treated as above. We can write  $X(z)$  as

$$X(z) = \sum_{k=0}^{M-N} B_k z^{-k} + \sum_{k=1}^N \frac{A_k}{1 - p_k z^{-1}}. \quad (2.130a)$$

The first summation of (2.130a) is clearly the  $z$ -transform of the sequence

$$\left[ \dots \ 0 \ \boxed{B_0} \ B_1 \ B_2 \ \dots \ B_{M-N} \ 0 \ \dots \right]^T.$$

There are many possible ROCs, each determining a distinct sequence corresponding to the second summation in (2.130a). When the ROC is outside of the largest pole, putting together both summations of (2.130a) yields

$$x_n = \sum_{k=0}^{M-N} B_k \delta_{n-k} + \sum_{k=1}^N A_k (p_k)^n. \quad (2.130b)$$

We illustrate the method with an example:

EXAMPLE 2.18 (INVERSION USING PARTIAL FRACTION EXPANSION) Given is

$$X(z) = \frac{1 - z^{-1}}{1 - 5z^{-1} + 6z^{-2}} = \frac{1 - z^{-1}}{(1 - 2z^{-1})(1 - 3z^{-1})},$$

with poles at  $z = 2$  and  $z = 3$ . We compute the coefficients as in (2.129a):

$$A_1 = \left. \frac{1 - z^{-1}}{1 - 3z^{-1}} \right|_{z=2} = -1, \quad A_2 = \left. \frac{1 - z^{-1}}{1 - 2z^{-1}} \right|_{z=3} = 2,$$

yielding

$$X(z) = \frac{-1}{1 - 2z^{-1}} + \frac{2}{1 - 3z^{-1}}.$$

The original sequence is then

$$x_n = \begin{cases} (2^n - 2 \cdot 3^n)u_{-n-1}, & \text{if ROC} = \{z \mid |z| < 2\}; \\ -2^n u_n - 2 \cdot 3^n u_{-n-1}, & \text{if ROC} = \{z \mid 2 < |z| < 3\}; \\ (-2^n + 2 \cdot 3^n)u_n, & \text{if ROC} = \{z \mid |z| > 3\}. \end{cases}$$

**Inversion Using Power-Series Expansion** This method is most useful for finite-length sequences. For example, given  $X(z) = (1 - z^{-1})(1 - 2z^{-1})$ , we can expand it in its power-series form as

$$X(z) = 1 - 3z^{-1} + 2z^{-2}.$$

Knowing that each of the elements in this power series corresponds to a delayed Kronecker delta sequence, we can read off the sequence directly:

$$x_n = \delta_n - 3\delta_{n-1} + 2\delta_{n-2}.$$

EXAMPLE 2.19 (INVERSION USING POWER-SERIES EXPANSION) Suppose

$$X(z) = \log(1 + 2z^{-1}), \quad \text{with ROC} = \{z \mid |z| > 2\}. \quad (2.131)$$

To invert this  $z$ -transform, we use its power-series expansion from Table P1.65-1. Substituting  $x = 2z^{-1}$ , we confirm that  $|x| = |2z^{-1}| < 2 \cdot \frac{1}{2} = 1$ , and thus the series expansion

$$\log(1 + 2z^{-1}) = \sum_{n=1}^{\infty} (-1)^{n+1} \frac{2^n}{n} z^{-n}$$

holds for the  $z$  values of interest. Thus, the desired inverse  $z$ -transform is

$$x_n = \begin{cases} (-1)^{n+1} 2^n n^{-1}, & \text{for } n \geq 1; \\ 0, & \text{otherwise.} \end{cases}$$

### 2.5.3 Properties of the $z$ -Transform

The  $z$ -transform has the same properties as the DTFT, but for a larger class of sequences. The main new twist is to properly account for ROCs. As an example, the convolution of two sequences can be computed as a product in the transform domain even when the sequences do not have proper DTFTs, provided that the sequences have some part of their ROCs in common. A summary of  $z$ -transform properties can be found in Table 2.6. As convolution in frequency as well as Parseval's equality involve contour integration, we opt not to state them here; a number of standard texts cover those.

**Linearity** The  $z$ -transform is a linear operator, or,

$$\alpha x_n + \beta y_n \xleftrightarrow{\text{ZT}} \alpha X(z) + \beta Y(z), \quad \text{ROC}_{\alpha x + \beta y} \supset \text{ROC}_x \cap \text{ROC}_y. \quad (2.132)$$

**Shift in Time** The  $z$ -transform pair corresponding to a shift in time by  $n_0$  is

$$x_{n-n_0} \xleftrightarrow{\text{ZT}} z^{-n_0} X(z), \quad (2.133)$$

with no changes to the ROC except possibly at  $z = 0$  or  $|z| = \infty$ .

**Scaling in Time** Scaling in time appears in two flavors:

- (i) The  $z$ -transform pair corresponding to scaling in time by  $N$  is

$$x_{Nn} \xleftrightarrow{\text{ZT}} \frac{1}{N} \sum_{k=0}^{N-1} X(W_N^k z^{1/N}), \quad (\text{ROC}_x)^{1/N}. \quad (2.134)$$

We have already seen this operation of downsampling in (2.87) and will discuss it in more detail in Section 2.7.

- (ii) The  $z$ -transform pair corresponding to scaling in time by  $1/N$  is

$$\begin{cases} x_{n/N}, & \text{for } n = \ell N, \quad \ell \in \mathbb{Z}; \\ 0, & \text{otherwise,} \end{cases} \xleftrightarrow{\text{ZT}} X(z^N), \quad (\text{ROC}_x)^N. \quad (2.135)$$

We have already seen this operation of upsampling in (2.88) and will discuss it in more detail in Section 2.7.

**Scaling in  $z$**  The  $z$ -transform pair corresponding to scaling in  $z$  by  $\alpha^{-1}$  is

$$\alpha^n x_n \xleftrightarrow{\text{ZT}} X(\alpha^{-1} z), \quad |\alpha| \text{ ROC}_x. \quad (2.136)$$

**Time Reversal** The  $z$ -transform pair corresponding to a time reversal  $x_{-n}$  is

$$x_{-n} \xleftrightarrow{\text{ZT}} X(z^{-1}), \quad \frac{1}{\text{ROC}_x}. \quad (2.137)$$

**Differentiation** The  $z$ -transform pair corresponding to differentiation in  $z$  is

$$n^k x_n \xleftrightarrow{\text{ZT}} (-1)^k z^k \frac{\partial^k X(z)}{\partial z^k}, \quad \text{ROC}_x. \quad (2.138)$$

**Moments** Computing the  $n$ th moment using the  $z$ -transform results in

$$m_k = \sum_{n \in \mathbb{Z}} n^k x_n = \left( \sum_{n \in \mathbb{Z}} n^k x_n z^{-n} \right) \Big|_{z=1} = (-1)^k \frac{\partial^k X(z)}{\partial z^k} \Big|_{z=1}, \quad k \in \mathbb{N}, \quad (2.139a)$$

as a direct application of (2.138). The first two moments are:

$$m_0 = \sum_{n \in \mathbb{Z}} x_n = \left( \sum_{n \in \mathbb{Z}} x_n z^{-n} \right) \Big|_{z=1} = X(0), \quad (2.139b)$$

$$m_1 = \sum_{n \in \mathbb{Z}} n x_n = \left( \sum_{n \in \mathbb{Z}} n x_n z^{-n} \right) \Big|_{z=1} = - \frac{\partial X(z)}{\partial z} \Big|_{z=1}. \quad (2.139c)$$

**Convolution in Time** The  $z$ -transform pair corresponding to convolution in time is

$$(h * x)_n \xleftrightarrow{\text{ZT}} H(z) X(z), \quad \text{ROC}_{h*x} \supset \text{ROC}_h \cap \text{ROC}_x. \quad (2.140)$$

This key result is the  $z$ -transform analogue of DTFT property (2.92). The  $z$ -transform of  $y = h * x$  can be obtained with slight modifications of (2.93):

$$\begin{aligned} Y(z) &\stackrel{(a)}{=} \sum_{n \in \mathbb{Z}} y_n z^{-n} \stackrel{(b)}{=} \sum_{n \in \mathbb{Z}} \left( \sum_{k \in \mathbb{Z}} x_k h_{n-k} \right) z^{-n} \\ &= \sum_{n \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} x_k z^{-k} h_{n-k} z^{-(n-k)} \\ &\stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} x_k z^{-k} \sum_{n \in \mathbb{Z}} h_{n-k} z^{-(n-k)} \stackrel{(d)}{=} X(z) H(z), \end{aligned} \quad (2.141)$$

where (a) follows from the definition of the  $z$ -transform; (b) from the definition of convolution; (c) from interchanging the order of summation; and (d) from the definition of the  $z$ -transform. The distinction from (2.93) is that (c) may hold when DTFTs of  $x$  and  $h$  do not exist; when  $z \in \text{ROC}_h \cap \text{ROC}_x$ , each series following (c) is absolutely convergent, enabling the interchange. The wider applicability of (2.140) than (2.92) is a key feature of the  $z$ -transform.

**EXAMPLE 2.20** ( $z$ -TRANSFORM CONVOLUTION PROPERTY) For some  $\alpha \in \mathbb{R}^+$ , consider

$$x_n = u_n, \quad h_n = \alpha^n u_n.$$

We cannot use the DTFT to compute the convolution  $y = h * x$  because  $x$  does not have a DTFT, and for  $\alpha \geq 1$ , neither does  $h$ . The  $z$ -transform exists for

<b><i>z</i>-transform properties</b>	<b>Time domain</b>	<b><i>z</i> domain</b>	<b>ROC</b>
<b>ROC properties</b>	General seq. Finite-length seq. Right-sided seq. Left-sided seq. BIBO stable		$0 \leq r_1 <  z  < r_2 \leq \infty$ all $z$ , except possibly $0, \infty$ $ z  > \text{largest pole}$ $ z  < \text{smallest pole}$ $\supset  z  = 1$
<b>Basic properties</b>			
Linearity	$\alpha x_n + \beta y_n$	$\alpha X(z) + \beta Y(z)$	$\supset \text{ROC}_x \cap \text{ROC}_y$
Shift in time	$x_{n-n_0}$	$z^{-n_0} X(z)$	$\text{ROC}_x$
Scaling in time			
Downsampling	$x_{Nn}$	$(1/N) \sum_{k=0}^{N-1} X(W_N^k z^{1/N})$	$(\text{ROC}_x)^{1/N}$
Upsampling	$x_{n/N}, n = \ell N; 0, \text{ otherwise}$	$X(z^N)$	$(\text{ROC}_x)^N$
Scaling in $z$	$\alpha^n x_n$	$X(\alpha^{-1} z)$	$ \alpha  \text{ROC}_x$
Time reversal	$x_{-n}$	$X(z^{-1})$	$1/\text{ROC}_x$
Differentiation	$n^k x_n$ $m_k = \sum_{n \in \mathbb{Z}} n^k x_n = (-1)^k \partial^k X(z) / \partial z^k \Big _{z=1}$	$(-1)^k z^k \partial^k X(z) / \partial z^k$	$\text{ROC}_x$
Convolution in time	$(h * x)_n$	$H(z) X(z)$	$\supset \text{ROC}_h \cap \text{ROC}_x$
Deterministic autocorrelation	$a_n = \sum_{k \in \mathbb{Z}} x_k x_{k-n}^*$	$A(z) = X(z) X_*(z^{-1})$	$\text{ROC}_x \cap 1/\text{ROC}_x$
Deterministic crosscorrelation	$c_n = \sum_{k \in \mathbb{Z}} x_k y_{k-n}^*$	$C(z) = X(z) Y_*(z^{-1})$	$1/\text{ROC}_x \cap \text{ROC}_y$
<b>Symmetries</b>			
Conjugate	$x_n^*$	$X^*(z^*)$	$\text{ROC}_x$
Conjugate, time reversed	$x_{-n}^*$	$X_*(z^{-1})$	$1/\text{ROC}_x$
Real part	$\Re(x_n)$	$(X(z) + X^*(z^*)) / 2$	$\text{ROC}_x$
Imaginary part	$\Im(x_n)$	$(X(z) - X^*(z^*)) / 2j$	$\text{ROC}_x$
$X$ conjugate symmetric	$x_n \text{ real}$	$X(z) = X^*(z^*)$	$\text{ROC}_x$
<b>Common transform pairs</b>			
Kronecker delta sequence	$\delta_n$	1	all $z$
Shift by $k$	$\delta_{n-k}$	$z^{-k}$	all $z$ , except possibly $0, \infty$
Exponential sequence	$\alpha^n u_n$ $-\alpha^n u_{-n-1}$	$1/(1 - \alpha z^{-1})$	$ z  >  \alpha $ $ z  <  \alpha $
Differentiation	$n \alpha^n u_n$ $-n, \alpha^n u_{-n-1}$	$(\alpha z^{-1}) / (1 - \alpha z^{-1})^2$	$ z  >  \alpha $ $ z  <  \alpha $

**Table 2.6:** Properties of the  $z$ -transform.

both  $x$  and  $h$ , and it can be used to compute the convolution provided that the ROCs overlap. The  $z$ -transforms are

$$X(z) = \frac{1}{1 - z^{-1}}, \quad \text{ROC}_x = \{z \mid |z| > 1\},$$

$$H(z) = \frac{1}{1 - \alpha z^{-1}}, \quad \text{ROC}_h = \{z \mid |z| > \alpha\},$$



and thus

$$Y(z) = \frac{1}{(1 - \alpha z^{-1})(1 - z^{-1})}, \quad \text{ROC}_y \supset \{z \mid |z| > \max\{\alpha, 1\}\}.$$

By partial fraction expansion, we can rewrite  $Y(z)$  as

$$Y(z) = \frac{-\alpha/(1 - \alpha)}{(1 - \alpha z^{-1})} + \frac{1/(1 - \alpha)}{(1 - z^{-1})},$$

leading to

$$y_n = -\frac{\alpha}{1 - \alpha} \alpha^n u_n + \frac{1}{1 - \alpha} u_n = \frac{1 - \alpha^{n+1}}{1 - \alpha} u_n.$$

As a check, we can compute the time-domain convolution directly:

$$y_n = \sum_{k \in \mathbb{Z}} x_k h_{n-k} = \sum_{k=0}^{\infty} h_{n-k} = \sum_{k=0}^n \alpha^{n-k} = \frac{1 - \alpha^{n+1}}{1 - \alpha} u_n.$$

When  $\alpha \in [0, 1)$ , the DTFT of  $y$  exists, but we nevertheless needed the  $z$ -transform to compute the convolution because the DTFT of  $x$  does not exist. When  $\alpha > 1$ , the DTFT of  $y$  does not exist, while the  $z$ -transform  $Y(z)$  exists with ROC  $\{z \mid |z| > \alpha\}$ .

**EXAMPLE 2.21 (FAILURE OF  $z$ -TRANSFORM CONVOLUTION PROPERTY)** Here is an example where the convolution sum converges, but even the  $z$ -transform does not help in computing it:

$$x_n = 1, \quad n \in \mathbb{Z}, \quad h_n = \alpha^n u_n, \quad 0 < \alpha < 1.$$

We can compute the convolution directly

$$y_n = h * x = \sum_{n \in \mathbb{Z}} h_n x_{k-n} = \sum_{n \in \mathbb{N}} \alpha^n = \frac{1}{1 - \alpha}.$$

However, there are no values of  $z$  such that the  $z$ -transform of  $x$  converges, that is, the ROC is empty. This prohibits the use of the  $z$ -transform for the computation of this convolution.

For finite-length, right-sided sequences, (2.140) connects convolution with polynomial multiplication. Given a length- $N$  sequence  $x$  and a length- $M$  impulse response  $h$ , the  $z$ -transforms of  $x$  and  $h$  are

$$H(z) = \sum_{n=0}^{M-1} h_n z^{-n}, \quad X(z) = \sum_{n=0}^{N-1} x_n z^{-n}.$$

Each is a polynomial in  $z^{-1}$ . The product polynomial  $H(z)X(z)$  has powers of  $z^{-1}$  going from 0 to  $M + N - 2$ , and its  $n$ th coefficient is obtained from the coefficients in  $H(z)$  and  $X(z)$  that have powers summing to  $n$ , that is, the convolution  $h * x$  given in (2.59).

**Deterministic Autocorrelation** The  $z$ -transform pair corresponding to the deterministic autocorrelation of a sequence  $x$  is

$$a_n = \sum_{k \in \mathbb{Z}} x_k x_{k-n}^* \xleftrightarrow{\text{ZT}} A(z) = X(z) X_*(z^{-1}), \quad \text{ROC}_x \cap \frac{1}{\text{ROC}_x}, \quad (2.142)$$

where  $X_*(z)$  denotes  $X^*(z^*)$ , which amounts to conjugating coefficients but not  $z$ . This  $z$ -transform satisfies

$$A(z) = A_*(z^{-1}). \quad (2.143a)$$

For a real  $x$ ,

$$A(z) = X(z) X(z^{-1}) = A(z^{-1}). \quad (2.143b)$$

The proof of (2.143b) is left for Exercise 2.13. We know that on the unit circle, the deterministic autocorrelation is the square magnitude of the spectrum  $|X(e^{j\omega})|^2$  as in (2.96). This quadratic form, when extended to the  $z$ -plane, leads to a particular symmetry of poles and zeros when  $X(z)$ , and thus  $A(z)$  as well, are rational functions.

**THEOREM 2.13 (RATIONAL AUTOCORRELATION)** A rational function  $A(z)$  is the  $z$ -transform of the deterministic autocorrelation of a stable real sequence  $x$ , if and only if:

- (i) its complex poles and zeros appear in quadruples:

$$\{z_i, z_i^*, z_i^{-1}, (z_i^{-1})^*\}, \quad \{p_i, p_i^*, p_i^{-1}, (p_i^{-1})^*\}; \quad (2.144a)$$

- (ii) its real poles and zeros appear in pairs:

$$\{z_i, z_i^{-1}\}, \quad \{p_i, p_i^{-1}\}; \quad (2.144b)$$

and

- (iii) its zeros on the unit circle are double zeros:

$$\{z_i, z_i^*, z_i^{-1}, (z_i^{-1})^*\} = \{e^{j\omega_i}, e^{-j\omega_i}, e^{-j\omega_i}, e^{j\omega_i}\}, \quad (2.144c)$$

with possibly double zeros at  $z = \pm 1$ . There are no poles on the unit circle.

*Proof.* The proof follows from the following two facts:

1.  $a_n$  is real, since  $x_n$  is real. From Table 2.6, this means that:

$$A^*(z) = A(z^*) \quad \Rightarrow \quad \begin{array}{ll} p_i \text{ pole} & \Rightarrow p_i^* \text{ pole} \\ z_i \text{ zero} & \Rightarrow z_i^* \text{ zero} \end{array} \quad (2.145a)$$

2.  $a_n$  is symmetric, since  $a_{-n} = a_n$ . From Table 2.6, this means that:

$$A(z^{-1}) = A(z) \quad \Rightarrow \quad \begin{array}{ll} p_i \text{ pole} & \Rightarrow p_i^{-1} \text{ pole} \\ z_i \text{ zero} & \Rightarrow z_i^{-1} \text{ zero} \end{array} \quad (2.145b)$$

We now proceed to prove that  $A(z)$  being the  $z$ -transform of the autocorrelation of a stable and real sequence  $x$  implies (i)-(iii). The converse follows similarly.

(i) From (2.145a)–(2.145b), we have that

$$p_i \text{ pole} \Rightarrow \begin{matrix} p_i^* & \text{pole} \\ p_i^{-1} & \text{pole} \end{matrix} \Rightarrow (p_i^*)^{-1} \text{ pole},$$

similarly for zeros, and we obtain the pole/zero quadruples in (2.144a).

- (ii) If a zero/pole is real, it is its own conjugate, and thus, quadruples in (2.144a) become pairs in (2.144b).
- (iii) Since  $x$  is stable, there are no poles on the unit circle. Since  $x$  is real,  $X^*(z) = X(z^*)$ . Thus, a rational  $A(z)$  has only zeros on the unit circle from  $X(z)$  and  $X(z^{-1})$ .

$$\begin{aligned} z_i \text{ zero of } X(z) &\Rightarrow \begin{matrix} z_i^{-1} & \text{zero of } X(z^{-1}) \\ z_i^* & \text{zero of } X(z) \end{matrix} \\ &\Rightarrow (z_i^*)^{-1} \text{ zero of } X(z^{-1}) \Rightarrow z_i = (z_i^*)^{-1} \text{ zero of } X(z). \end{aligned}$$

Thus, both  $X(z)$  and  $X(z^{-1})$  have  $z_i$  as a zero, leading to double zeros on the unit circle.

**Deterministic Crosscorrelation** The  $z$ -transform pair corresponding to the deterministic crosscorrelation of sequences  $x$  and  $y$  is

$$c_n = \sum_{k \in \mathbb{Z}} x_k y_{k-n}^* \xleftrightarrow{\text{ZT}} C_{x,y}(z) = X(z) Y_*(z^{-1}), \quad \frac{1}{\text{ROC}_x} \cap \text{ROC}_y, \quad (2.146)$$

and satisfies

$$C_{x,y}(z) = C_{y,x^*}(z^{-1}). \quad (2.147a)$$

For  $x, y$  real,

$$C_{x,y}(z) = X(z) Y(z^{-1}) = C_{y,x}(z^{-1}). \quad (2.147b)$$

**Deterministic Autocorrelation of Vector Sequences** The  $z$ -transform pair corresponding to the deterministic autocorrelation of a vector sequence  $x$  is

$$A_n \xleftrightarrow{\text{ZT}} A(z) = \begin{bmatrix} A_0(z) & C_{0,1}(z) & \dots & C_{0,N-1}(z) \\ C_{1,0}(z) & A_1(z) & \dots & C_{1,N-1}(z) \\ \vdots & \vdots & \ddots & \vdots \\ C_{N-1,0}(z) & C_{N-1,1}(z) & \dots & A_{N-1}(z) \end{bmatrix}, \quad (2.148)$$

where  $A_n$  is given in (2.22). Because of (2.20a),  $A(z)$  satisfies

$$A(z) = \begin{bmatrix} A_0(z) & C_{0,1}(z) & \dots & C_{0,N-1}(z) \\ C_{0,1^*}(z^{-1}) & A_1(z) & \dots & C_{1,N-1}(z) \\ \vdots & \vdots & \ddots & \vdots \\ C_{0,N-1^*}(z^{-1}) & C_{1,N-1^*}(z^{-1}) & \dots & A_{N-1}(z) \end{bmatrix} = A_*(z^{-1}). \quad (2.149a)$$

Here,  $A_*(z) = A^*(z^*)$  extends the previous notation to mean transposition of  $A$  and conjugation of coefficients, but not of  $z$ .<sup>52</sup> For a real  $x$ ,

$$A(z) = A^T(z^{-1}). \quad (2.149b)$$

**Spectral Factorization** The particular pattern of poles and zeros which characterizes a rational autocorrelation in Theorem 2.13 leads to a key procedure called *spectral factorization*. This amounts to taking the square root of  $A(e^{j\omega})$ , and by extension, of  $A(z)$ , factoring it into rational factors  $X(z)$  and  $X(z^{-1})$ ,<sup>53</sup> as a direct corollary of Theorem 2.13.

**COROLLARY 2.14 (SPECTRAL FACTORIZATION)** A rational  $z$ -transform  $A(z)$  is the deterministic autocorrelation of a stable real sequence  $x_n$  if and only if it can be factored as  $A(z) = X(z)X(z^{-1})$ .

Spectral factorization amounts to assigning poles and zeros from quadruples and pairs (2.144a)–(2.144c) to  $X(z)$  and  $X(z^{-1})$ . For the poles, there is a unique rule: take all poles inside the unit circle and assign them to  $X(z)$ . This is because stability of  $x$  requires  $X(z)$  to have only poles inside the unit circle (Proposition 2.15), while  $x$  real requires that conjugate pairs be kept together. For the zeros, there is a choice, since we are not forced to only assign zeros inside the unit circle to  $X(z)$ . Doing so, however, creates a unique solution called the *minimum-phase solution*.<sup>54</sup> It is now clear why it is important that the zeros on the unit circle appear in pairs: it allows for the assignment of one of each to  $X(z)$  and  $X(z^{-1})$ .

**EXAMPLE 2.22 (SPECTRAL FACTORIZATION)** We now illustrate both the procedure and how we can recognize a deterministic autocorrelation of a real and stable sequence (see Figure 2.14):

- (i) The first sequence we examine is a finite-length, symmetric sequence  $a_n$  with its associated  $z$ -transform:

$$\begin{aligned} a_n &= 2\delta_{n+1} + 5\delta_n + 2\delta_{n-1}, \\ A(z) &= 5 + 2(z + z^{-1}) = (1 + 2z^{-1})(1 + 2z), \end{aligned}$$

depicted in Figure 2.14(a). This sequence is a deterministic autocorrelation since it has two zeros and they appear in a pair as per Theorem 2.13. As we said above, we have a choice whether to assign  $-\frac{1}{2}$  or  $-2$  to  $X(z)$ ; the minimum-phase solution assigns  $-\frac{1}{2}$  to  $X(z)$  and  $-2 = (-\frac{1}{2})^{-1}$  to  $X(z^{-1})$ .

<sup>52</sup>Note that in (2.102) we could have written the elements below the diagonal, for example,  $C_{0,1}^*(e^{j\omega})$ , as  $C_{0,1*}(e^{-j\omega})$  to parallel the  $z$ -transform. Here, subscript  $*$  would just mean conjugation of coefficients, as conjugation of  $e^{j\omega}$  is taken care of by negation.

<sup>53</sup>Note that since  $A(e^{j\omega})$  is real and nonnegative, one could write  $X(e^{j\omega}) = \sqrt{A(e^{j\omega})}$ . However, such a spectral root will in general not be rational.

<sup>54</sup>The name stems from the fact that among the various solutions, this one will create a minimal delay, or, that the sequence is most concentrated towards the origin of time.

- (ii) The second sequence is an infinite-length, symmetric sequence  $a_n$  with its associated  $z$ -transform:

$$a_n = \left(\frac{1}{2}\right)^n u_n + 2^n u_{-n-1},$$

$$A(z) = \frac{1}{1 - \frac{1}{2}z^{-1}} - \frac{1}{1 - 2z^{-1}} = -\frac{(3/2)z^{-1}}{(1 - \frac{1}{2}z^{-1})(1 - 2z^{-1})},$$

depicted in Figure 2.14(b). Above, we have used the  $z$ -transform pairs from Table 2.6. This sequence is a deterministic autocorrelation since it has two poles and they appear in a pair as per Theorem 2.13. We now have no choice but to assign  $\frac{1}{2}$  to  $X(z)$ , as for a stable sequence all its poles must be inside the unit circle; the other pole,  $2 = (\frac{1}{2})^{-1}$  goes to  $X(z^{-1})$ .

- (iii) Finally, we examine the following finite-length, symmetric sequence  $a_n$  with its associated  $z$ -transform:

$$a_n = 2\delta_{n+1} + 7\delta_n + 7\delta_{n-1} + 2\delta_{n-2},$$

$$A(z) = 7(1 + z^{-1}) + 2(z + z^{-2}) = (1 + \frac{1}{2}z^{-1})(1 + 2z^{-1})(1 + z^{-1}),$$

depicted in Figure 2.14(c). This sequence is not a deterministic autocorrelation since it has three zeros, two appearing in a pair as in Part (i) and the third, a single zero on the unit circle, violating Theorem 2.13. The DTFT of  $a_n$  is not real, for example,  $A(e^{j\pi/2}) = -(5/2)(1 + j)$ .

### 2.5.4 $z$ -Transform of Filters

For filters,

$$H(z) = \sum_{n \in \mathbb{Z}} h_n z^{-n} \quad (2.150)$$

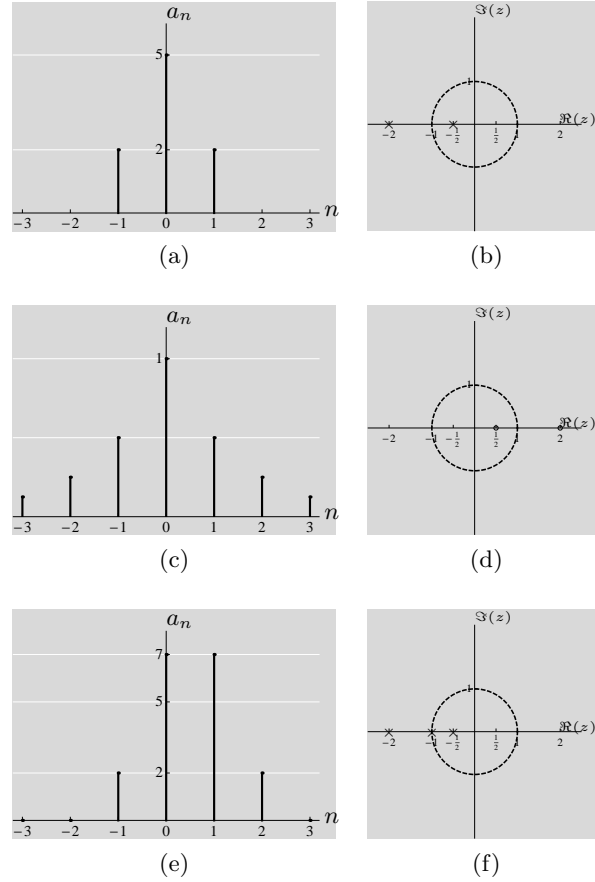
is the counterpart of the frequency response in (2.105a); it is well defined for values of  $z$  for which  $h_n z^{-n}$  is absolutely summable. As mentioned previously, there is a one-to-one relationship between a rational  $z$ -transform and a realizable difference equation (one with a finite number of coefficients). After revisiting this relationship, we establish a necessary and sufficient condition for stability of causal systems with rational transfer functions.

**Difference Equations with Finite Number of Coefficients** Consider a causal solution of a difference equation with a finite number of terms as in (2.54) with zero initial conditions. Assuming  $x$  and  $y$  have well-defined  $z$ -transforms  $X(z)$  and  $Y(z)$ , and using that  $x_{n-k}$  and  $z^{-k}X(z)$  are a  $z$ -transform pair, we can rewrite (2.54) as

$$Y(z) = \left( \sum_{k=0}^M b_k z^{-k} \right) X(z) - \left( \sum_{k=1}^N a_k z^{-k} \right) Y(z).$$

The *transfer function* is given by

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}}. \quad (2.151)$$



**Figure 2.14:** Pole/zero locations of rational autocorrelations. (a) Finite-length, symmetric sequence that is an autocorrelation and its (b) zero locations. (c) Infinite-length, symmetric sequence that is a deterministic autocorrelation and its (d) pole locations. (e) Finite-length, symmetric sequence that is not a deterministic autocorrelation and its (f) zero locations.

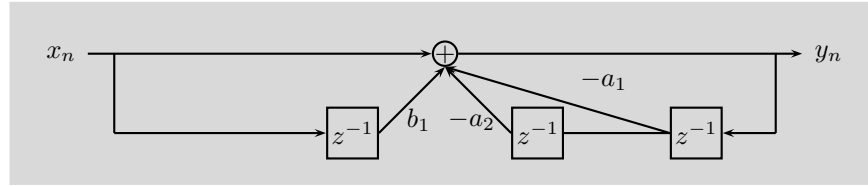
That is, a linear discrete-time system satisfying difference equation (2.54) has a rational transfer function  $H(z)$  in the  $z$ -transform domain; in other words, the  $z$ -transform of the impulse response of the system is a rational function.

**EXAMPLE 2.23 (RATIONAL TRANSFER FUNCTION)** Consider the simple system in Figure 2.15. The constraint at the summing node gives

$$y_n = x_n + b_1 x_{n-1} - a_1 y_{n-1} - a_2 y_{n-2}.$$

By using  $z$ -transform properties, this implies

$$Y(z) = X(z) + b_1 z^{-1} X(z) - a_1 z^{-1} Y(z) - a_2 z^{-2} Y(z).$$



**Figure 2.15:** A simple discrete-time system, where  $z^{-1}$  stands for a unit delay.

The system transfer function is therefore

$$H(z) = \frac{1 + b_1 z^{-1}}{1 + a_1 z^{-1} + a_2 z^{-2}}.$$

We now discuss stability for systems with rational transfer functions.

**PROPOSITION 2.15 (STABILITY)** A causal, LSI discrete-time system with a rational transfer function is BIBO stable if and only if the poles of its (reduced) transfer function are inside the unit circle.

*Proof.* Using the partial fraction inversion method described in Section 2.5.2, the impulse response of a causal, LSI discrete-time system with rational transfer function is a linear combination of right-sided geometric sequences as in (2.129b)—possibly with multiplication by  $n^k$  factors (stemming from multiplicities of poles) and with additional terms that are shifted Kronecker delta sequences (from the numerator having higher degree than the denominator as in (2.130b)). When each pole is inside the unit circle, each term in this linear combination is absolutely summable, so the impulse response is absolutely summable as well; thus, according to Proposition 2.8, the system is BIBO stable. Conversely, if any pole is outside the unit circle, the impulse response is not absolutely summable; thus, according to Proposition 2.8, the system is not BIBO stable.

## Filters

A major application of the  $z$ -transform is in the analysis and design of filters. With the restriction to rational functions for realizability, designing a desirable filter is essentially the problem of strategically placing poles and zeros in the  $z$ -plane. As simple as this may sound, filter design is a rather sophisticated problem, and it has led to a vast literature and numerous numerical procedures. For example, a standard way to design FIR filters is to use a numerical optimization procedure such as the Parks–McClellan algorithm, which iteratively modifies coefficients so as to approach a desired frequency response. Rather than embarking on a tour of filter design, we study properties of certain classes of filters; pointers to filter design techniques are given in *Further Reading*.

**FIR Filters** The  $z$ -transform of a length- $L$  FIR filter is a polynomial in  $z^{-1}$ ,

$$H(z) = \sum_{n=0}^{L-1} h_n z^{-n},$$

and is given in its factored form as (2.126).

**Linear-Phase Filters** In  $z$ -domain, the symmetries from (2.110) become:

$$\begin{array}{ll} \text{symmetric} & \xleftrightarrow{\text{ZT}} H(z) = z^{-L+1} H(z^{-1}), \end{array} \quad (2.152a)$$

$$\begin{array}{ll} \text{antisymmetric} & \xleftrightarrow{\text{ZT}} H(z) = -z^{-L+1} H(z^{-1}), \end{array} \quad (2.152b)$$

In  $z$ -domain,  $H(z^{-1})$  reverses the filter,  $z^{-L+1}$  makes it causal again, and  $\pm$  determines the type of symmetry.

**Allpass Filters** The basic single-zero/single-pole allpass building block given in (2.115) has the  $z$ -transform

$$H(z) = \frac{z^{-1} - \alpha^*}{1 - \alpha z^{-1}}, \quad (2.153)$$

with the zero  $1/\alpha^*$  and pole  $\alpha$ . For stability in the causal case,  $|\alpha| < 1$  is required. A more general allpass filter is formed by cascading these elementary building blocks as

$$H(z) = \prod_{i=1}^N \frac{z^{-1} - \alpha_i^*}{1 - \alpha_i z^{-1}} = z^{-N} \frac{B_*(z^{-1})}{B(z)}, \quad (2.154)$$

where  $B(z) = \prod_{i=1}^N (1 - \alpha_i z^{-1})$ . The  $z$ -transform of the deterministic autocorrelation of such an allpass filter is given by

$$A(z) = H(z)H_*(z^{-1}) = \prod_{i=1}^N \frac{z^{-1} - \alpha_i^*}{1 - \alpha_i z^{-1}} \prod_{i=1}^N \frac{z - \alpha_i}{1 - \alpha_i^* z} = 1,$$

and thus, an allpass filter has the deterministic autocorrelation sequence  $a_m = \delta_m$ . Poles and zeros appear in pairs as  $\{\alpha, 1/\alpha^*\} = \{r_0 e^{j\omega_0}, (1/r_0) e^{j\omega_0}\}$  for some real  $r_0 \in (0, 1)$  and angle  $\omega_0$ . They appear across the unit circle at the same angle and at reciprocal magnitudes, and thus the magnitude  $|H(e^{j\omega})|$  is not influenced while the phase is, as was shown in Figure 2.12.

## 2.6 Discrete Fourier Transform

As mentioned previously in this chapter, one way in which a finite-length sequence arises is as one period of an infinite-length periodic sequence. The version of the



Fourier transform designed for finite-length sequences treats all finite-length sequences this way, so effectively we are circularly extending any finite-length sequences. As we have seen in Section 2.3.3, the circular convolution operator (2.69) is the appropriate description of LSI systems operating on finite-length inputs, circularly extended.

The version of the Fourier transform for this combination of sequence space and convolution is the discrete Fourier transform. Similarly to our discussion on eigensequences of the linear convolution operator leading to the definition of the DTFT, we will find appropriate eigensequences of the circular convolution operator leading to the DFT. As expected from this construction, the DFT diagonalizes the circular convolution operator.

One of the most important uses of the DFT and circular convolution arises from their connections with the DTFT and linear convolution. Specifically, we will see that the DFT is a tool for fast computation of linear convolutions.

### 2.6.1 Definition of the DFT

**Eigensequences of the Circular Convolution Operator** Given that we have an appropriate convolution operator defined (circular convolution in (2.69)), mimicking what we did for the DTFT, the DFT arises from identifying the eigensequences of the convolution operator. We can guess that the eigensequences are of unit-modulus complex exponential form  $v_n = e^{j\omega n}$  like in Section 2.4.1. In addition, since we will represent sequences of period  $N$  with these eigensequences, we can guess that the eigensequences should be periodic with period  $N$  as well. Due to the periodicity,

$$v_{n+N} = e^{j\omega(n+N)} = v_n \Rightarrow e^{j\omega N} = 1 \Rightarrow \omega = \frac{2\pi}{N}k, \quad (2.155)$$

for  $k \in \mathbb{Z}$ . Since  $k$  and  $k + \ell N$  lead to the same complex exponential sequence, we have  $N$  distinct complex exponential sequences of period  $N$ , indexed by  $k \in \{0, 1, \dots, N-1\}$  instead of  $\omega \in \mathbb{R}$ :

$$v_n = e^{j(2\pi/N)kn} = W_N^{-kn}, \quad v = \begin{bmatrix} 1 & W_N^{-k} & \dots & W_N^{-(N-1)k} \end{bmatrix}^T, \quad (2.156)$$

where  $W_N = e^{-j2\pi/N}$  (for details, see (2.276) in Appendix 2.A.1).

Let us check that these are indeed eigensequences of the circular convolution operator  $H$  from (2.69):

$$\begin{aligned} (Hv)_n &= (h \circledast v)_n = \sum_{i=0}^{N-1} v_{(n-i) \bmod N} h_i = \sum_{i=0}^{N-1} W_N^{k[(n-i) \bmod N]} h_i \\ &\stackrel{(a)}{=} \sum_{i=0}^{N-1} W_N^{k(n-i)} h_i = \underbrace{\sum_{i=0}^{N-1} h_i W_N^{-ki}}_{\lambda_k} \underbrace{W_N^{kn}}_{v_n}, \end{aligned} \quad (2.157)$$

where (a) follows from the fact that if  $(n-i) = \ell N + p$ , then  $(n-i) \bmod N = p$  and  $W_N^{[(n-i) \bmod N]} = W_N^p$ , but also  $W_N^{(n-i)} = W_N^{(\ell N + p)} = W_N^p$ . Thus indeed, applying

the convolution operator  $H$  to the complex exponential sequence  $v$  results in the same sequence, albeit scaled by the corresponding eigenvalue  $\lambda_k$ . As before, we call that eigenvalue the *frequency response* of the system  $H_k$ ; it is defined formally in (2.176a). We can thus rewrite (2.157) as

$$H W_N^{-kn} = h \otimes W_N^{-kn} = H_k W_N^{-kn}. \quad (2.158)$$

Each of the  $N$  distinct eigensequences generates the space  $S_k = \{\alpha e^{j(2\pi/N)kn} \mid \alpha \in \mathbb{C}\}$ . The quantity  $k$  is called *discrete frequency*.

**DFT** Finding the appropriate Fourier transform now amounts to projecting onto each of the invariant subspaces  $S_k$ .

**DEFINITION 2.16 (DISCRETE FOURIER TRANSFORM)** The discrete Fourier transform of a length- $N$  sequence  $x$  is

$$X_k = (F x)_k = \sum_{n=0}^{N-1} x_n W_N^{kn}, \quad k \in \{0, 1, \dots, N-1\}; \quad (2.159a)$$

we call it the *spectrum* of  $x$ . The inverse DFT of a length- $N$  sequence  $X$  is

$$x_n = \frac{1}{N} (F^* X)_k = \frac{1}{N} \sum_{k=0}^{N-1} X_k W_N^{-kn}, \quad n \in \{0, 1, \dots, N-1\}. \quad (2.159b)$$

We denote the DFT pair as

$$x_n \xleftrightarrow{\text{DFT}} X_k.$$

Within the definition, we have introduced  $F : \mathbb{C}^N \rightarrow \mathbb{C}^N$  to represent the linear DFT operator. The relationship between the inverse and the adjoint in (2.159b) is verified shortly.

**DFT as an Orthonormal Basis** If we define

$$\varphi_k = \frac{1}{\sqrt{N}} v_k = \frac{1}{\sqrt{N}} e^{j(2\pi/N)kn} = \frac{1}{\sqrt{N}} W_N^{-kn}, \quad (2.160)$$

it is easy to see from (2.159a)–(2.159b), as well as from orthogonality of  $v_k$  (easily checked using (2.277c)), that the set  $\{\varphi_k\}_{k=0}^{N-1}$  forms an orthonormal basis for  $\mathbb{C}^N$ .

**Matrix View** We know how to form a matrix given an orthonormal basis by stacking the basis vectors as its columns; however, as this is the community standard, the scaled basis vectors would go into the matrix  $F^*$  and not  $F$ . Thus, the DFT

## 2.6. Discrete Fourier Transform

243

operator  $F$  and its inverse  $F^{-1}$  are given by:

$$F = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_N & W_N^2 & \dots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \dots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{(N-1)} & W_N^{2(N-1)} & \dots & W_N^{(N-1)^2} \end{bmatrix}, \quad (2.161a)$$

$$F^{-1} = \frac{1}{N} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_N^{-1} & W_N^{-2} & \dots & W_N^{-(N-1)} \\ 1 & W_N^{-2} & W_N^{-4} & \dots & W_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{-(N-1)} & W_N^{-2(N-1)} & \dots & W_N^{-(N-1)^2} \end{bmatrix} = \frac{1}{N} F^*. \quad (2.161b)$$

This shows once more that the DFT is a unitary operator (up to a scaling factor).<sup>55</sup> Note that  $F$  is a Vandermonde matrix (see (1.230)), and its determinant is

$$\det(F) = \frac{1}{\sqrt{N}}, \quad \det(F^{-1}) = \sqrt{N}, \quad \det(F^*) = N^{N+1/2}. \quad (2.162)$$

**Relation of the DFT to the DTFT** When given a length- $N$  sequence  $x$  to analyze, we might first turn to what we already have—the DTFT. To turn it into a tool for analyzing length- $N$  sequences, we can simply sample it. As we need only  $N$  points, we can choose them anywhere; let us choose  $N$  uniformly spaced points in frequency given by (2.155). Then, evaluate the DTFT at these  $N$  distinct points:

$$\begin{aligned} X(e^{j\omega})|_{(2\pi/N)k} &\stackrel{(a)}{=} X(e^{j(2\pi/N)k}) \stackrel{(b)}{=} \sum_{n \in \mathbb{Z}} x_n e^{-j(2\pi/N)kn} \\ &\stackrel{(c)}{=} \sum_{n=0}^{N-1} x_n e^{-j(2\pi/N)kn} = X_k, \end{aligned}$$

where (a) follows from sampling the DTFT uniformly at  $\omega = 2\pi k/N$ ; (b) from the expression for the DTFT (2.78a); and (c) from  $x$  being finite of length  $N$ . The final expression is the one for the DFT we have seen in (2.159a). Thus, sampling the DTFT results in the DFT.

We can now easily find the appropriate convolution operator corresponding to the DFT (while we know what it is already, we reverse engineer it here). The way to do it is to ask ourselves which operator will have as its eigensequences the columns of the DFT matrix  $F^*$  (since then, that operator is diagonalized by the DFT). In other words, which operator  $H$  will satisfy the following:

$$H v_k = \lambda_k v_k \quad \Rightarrow \quad H F^* = F^* \Lambda \quad \Rightarrow \quad H = \frac{1}{N} F^* \Lambda F.$$

<sup>55</sup>A normalized version uses a  $1/\sqrt{N}$  on both  $F$  and its inverse.

Even though we do not know  $\Lambda$ , we know it is a diagonal matrix of eigenvalues  $\lambda_k$ . Using the expressions (2.161a)–(2.161b) for  $F$  and  $F^*$ , we can find the expression for the element  $H_{i,m}$  as

$$H_{i,m} = \sum_{\ell=0}^{N-1} \lambda_{\ell} W_N^{\ell(m-i)},$$

which shows  $H$  to be a circulant matrix (circular convolution). We leave the details of the derivation as exercise.

What we have seen in this short account is how to reverse engineer the circular convolution given a finite-length sequence and the DFT.

## 2.6.2 Properties of the DFT

We list here the basic properties of the DFT; Table 2.7 summarizes these, together with symmetries as well as standard transform pairs, while Exercise 2.19 explores proofs for some of the properties.

**Linearity** The DFT operator  $F$  is a linear operator, or,

$$\alpha x_n + \beta y_n \xleftrightarrow{\text{DFT}} \alpha X_k + \beta Y_k. \quad (2.163)$$

**Shift in Time** The DFT pair corresponding to a shift in time by  $n_0$  is

$$x_{(n-n_0) \bmod N} \xleftrightarrow{\text{DFT}} W_N^{kn_0} X_k. \quad (2.164)$$

**Shift in Frequency** The DFT pair corresponding to a shift in frequency by  $k_0$  is

$$W_N^{-k_0 n} x_n \xleftrightarrow{\text{DFT}} X_{(k-k_0) \bmod N}. \quad (2.165)$$

As for the DTFT, a shift in frequency is often referred to as *modulation in time*, and is dual to the shift in time.

**Time Reversal** The DFT pair corresponding to a time reversal  $x_{-n \bmod N}$  is

$$x_{-n \bmod N} \xleftrightarrow{\text{DFT}} X_{-k \bmod N}; \quad (2.166)$$

the proof is given in Exercise 2.19.

**Convolution in Time** The DFT pair corresponding to convolution in time is

$$(h \circledast x)_n \xleftrightarrow{\text{DFT}} H_k X_k. \quad (2.167)$$

As this is a central concept in signal processing, it is worth repeating the following: (1) given a finite-length sequence and a finite-length filter, their linear convolution can be computed using a circular convolution of appropriate length as in Proposition 2.10; and (2) the DFT operator  $F$  diagonalizes the circular convolution operator  $H$  as in (2.177).

## 2.6. Discrete Fourier Transform

245

DFT properties	Time domain	DFT domain
<b>Basic properties</b>		
Linearity	$\alpha x_n + \beta y_n$	$\alpha X_k + \beta Y_k$
Shift in time	$x_{(n-n_0) \bmod N}$	$W_N^{kn_0} X_k$
Shift in frequency	$W_N^{-k_0 n} x_n$	$X_{(k-k_0) \bmod N}$
Time reversal	$x_{-n \bmod N}$	$X_{-k \bmod N}$
Convolution in time	$(h \otimes x)_n$	$H_k X_k$
Convolution in frequency	$h_n x_n$	$(1/N)(H \otimes X)_k$
Deterministic autocorrelation	$a_n = \sum_{k=0}^{N-1} x_k x_{(k-n) \bmod N}^*$	$A_k =  X_k ^2$
Deterministic crosscorrelation	$c_n = \sum_{k=0}^{N-1} x_k y_{(k-n) \bmod N}^*$	$C_k = X_k Y_k^*$
Parseval's equality	$\ x\ ^2 = \sum_{n=0}^{N-1}  x_n ^2 = (1/N) \sum_{k=0}^{N-1}  X_k ^2 = (1/N) \ X\ ^2$	
<b>Symmetries</b>		
Conjugate	$x_n^*$	$X_{-k \bmod N}^*$
Conjugate, time reversed	$x_{-n \bmod N}^*$	$X_k^*$
Real part	$\Re(x_n)$	$(X_k + X_{-k \bmod N}^*)/2$
Imaginary part	$\Im(x_n)$	$(X_k - X_{-k \bmod N}^*)/2j$
Conjugate-symmetric part	$(x_n + x_{-n \bmod N}^*)/2$	$\Re(X_k)$
Conjugate-antisymmetric part	$(x_n - x_{-n \bmod N}^*)/2j$	$\Im(X_k)$
<b>Symmetries for real <math>x</math></b>		
$X$ conjugate symmetric		$X_k = X_{-k \bmod N}^*$
Real part of $X$ even		$\Re(X_k) = \Re(X_{-k \bmod N})$
Imaginary part of $X$ odd		$\Im(X_k) = -\Im(X_{-k \bmod N})$
Magnitude of $X$ even		$ X_k  =  X_{-k \bmod N} $
Phase of $X$ odd		$\arg X_k = -\arg X_{-k \bmod N}$
<b>Common transform pairs</b>		
Kronecker delta sequence	$\delta_n$	1
Shift by $n_0$	$\delta_{(n-n_0) \bmod N}$	$W_N^{kn_0}$
Exponential sequence	$\alpha^n$	$(1 - \alpha W_N^{kN}) / (1 - \alpha W_N^k)$
Ideal lowpass filter	$\sqrt{\frac{k_0}{N}} \frac{\text{sinc}(\pi n k_0 / N)}{\text{sinc}(\pi n / N)}$	$\begin{cases} \sqrt{\frac{N}{k_0}}, &  k - \frac{N}{2}  \geq \frac{k_0-1}{2}; \\ 0, & \text{otherwise.} \end{cases}$
Box sequence	$\begin{cases} \frac{1}{\sqrt{n_0}}, &  n - \frac{N}{2}  \geq \frac{n_0-1}{2}; \\ 0, & \text{otherwise.} \end{cases}$	$\sqrt{n_0} \frac{\text{sinc}(\pi n_0 k / N)}{\text{sinc}(\pi k / N)}$

Table 2.7: Properties of the DFT.

**Convolution in Frequency** The DFT pair corresponding to convolution in frequency is

$$h_n x_n \xrightarrow{\text{DFT}} \frac{1}{N} (H \otimes X)_k; \quad (2.168)$$

the proof is given in Exercise 2.19. As expected, convolution in frequency is dual to convolution in time.

**Deterministic Autocorrelation** The DFT pair corresponding to the deterministic autocorrelation of a sequence  $x$  is

$$a_n = \sum_{k=0}^{N-1} x_k x_{(k-n) \bmod N}^* \xleftrightarrow{\text{DFT}} A_k = |X_k|^2, \quad (2.169)$$

for  $n, k = 0, 1, \dots, N-1$ , and satisfies

$$A_k = A_{-k \bmod N}^*. \quad (2.170a)$$

For a real  $x$ ,

$$A_k = |X_k|^2 = A_{-k \bmod N}. \quad (2.170b)$$

**Deterministic Crosscorrelation** The DFT pair corresponding to the deterministic crosscorrelation of sequences  $x$  and  $y$  is

$$c_n = \sum_{k=0}^{N-1} x_k y_{(k-n) \bmod N}^* \xleftrightarrow{\text{DFT}} C_k = X_k Y_k^*, \quad (2.171)$$

and satisfies

$$C_{x,y,k} = C_{y,x,-k \bmod N}^*. \quad (2.172a)$$

For  $x, y$  real,

$$C_{x,y,k} = X_k Y_k = C_{y,x,-k \bmod N}. \quad (2.172b)$$

**Deterministic Autocorrelation of Vector Sequences** The DFT pair corresponding to the deterministic autocorrelation of a vector sequence  $x$  is

$$A_n \xleftrightarrow{\text{DFT}} A_k = \begin{bmatrix} A_{0,k} & C_{0,1,k} & \dots & C_{0,N-1,k} \\ C_{1,0,k} & A_{1,k} & \dots & C_{1,N-1,k} \\ \vdots & \vdots & \ddots & \vdots \\ C_{N-1,0,k} & C_{N-1,1,k} & \dots & A_{N-1,k} \end{bmatrix}, \quad (2.173)$$

and satisfies

$$A_k = \begin{bmatrix} A_{0,k} & C_{0,1,k} & \dots & C_{0,N-1,k} \\ C_{0,1,-k \bmod N}^* & A_{1,k} & \dots & C_{1,N-1,k} \\ \vdots & \vdots & \ddots & \vdots \\ C_{0,N-1,-k \bmod N}^* & C_{1,N-1,-k \bmod N}^* & \dots & A_{N-1,k} \end{bmatrix} = A_{-k \bmod N}^*. \quad (2.174a)$$

For a real vector of sequences  $x$ ,

$$A_k = A_{-k \bmod N}^T. \quad (2.174b)$$

**Parseval's Equality** The DFT operator  $F$  is a unitary operator (within scaling) and thus preserves the Euclidean norm (see (1.51)):

$$\|x\|^2 = \sum_{n=0}^{N-1} |x_n|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X_k|^2 = \frac{1}{N} \|X\|^2 = \frac{1}{N} \|Fx\|^2. \quad (2.175)$$

This follows from  $F/\sqrt{N}$  being a unitary matrix since  $F^*F = NI$ .

### 2.6.3 Frequency Response of Filters

As for the DTFT, the DFT is defined for sequences and we use spectrum to denote their DTFTs. The *frequency response* is defined for filters (systems) as

$$H_k = \sum_{n=0}^{N-1} h_n W_N^{kn}, \quad k \in \{0, 1, \dots, N-1\}, \quad (2.176a)$$

with the corresponding impulse response,

$$h_n = \frac{1}{N} \sum_{k=0}^{N-1} H_k W_N^{-kn}, \quad n \in \{0, 1, \dots, N-1\}. \quad (2.176b)$$

We can again denote the magnitude and phase as

$$H_k = |H_k| e^{j \arg(H_k)},$$

where  $|H_k|$  is an  $N$ -periodic real positive sequence—the *magnitude response*, and  $\arg(H_k)$  is an  $N$ -periodic, real sequence between 0 and  $N-1$ —the *phase response*. A filter is said to possess *linear phase* when the phase response is linear in  $k$ . A filter is called *bandlimited* when its frequency response is finitely supported.

**Diagonalization of the Circular Convolution Operator** Call  $\Lambda = \text{diag}([H_k])_{k=0}^{N-1}$ , with  $H_k$  the DFT coefficients of  $h_n$  as in (2.176a). Then, from (2.157), we can immediately see that the DFT operator  $F$  (2.161a) diagonalizes the circular convolution operator  $H$  (2.73) as in (1.210a),

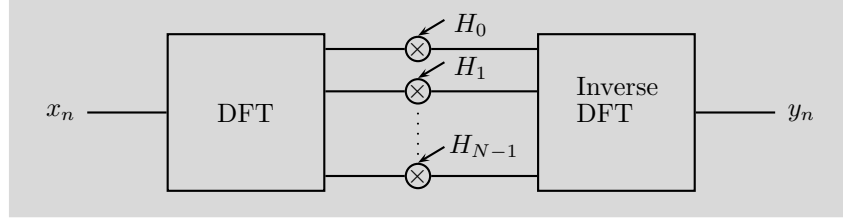
$$H = F \Lambda F^{-1}. \quad (2.177)$$

We illustrate this diagonalization in Figure 2.16) and explore it further in Solved Exercise 2.5.

**Ideal Filters** TBD.

**FIR Filters** TBD.

**Linear-Phase Filters** TBD.



**Figure 2.16:** Diagonalization property of the DFT.

**Allpass Filters** TBD.

## 2.7 Multirate Sequences and Systems

So far, we considered sequences in time indexed by integers, and the time index was assumed to be the same for all sequences. Physically, this is as if we had observed a physical process, and each term in the sequence corresponded to a sample of the process taken at regular intervals (for example, every second).

In multirate sequence processing, different sequences may have different time scales. Thus, the index  $n$  of the sequence may refer to different physical times for different sequences. We might ask both why do that and how to go between these different scales. Let us look at a simple example. Start with a sequence  $x_n$  and derive a downsampled sequence  $y_n$  by dropping every other sample

$$y_n = x_{2n} \rightarrow [\dots x_{-2} \boxed{x_0} x_2 \dots]^T, \quad (2.178)$$

Clearly, if  $x_n$  is the sample of a physical process taken at  $t = n$ , then  $y_n$  is a sample taken at time  $t = 2n$ . In other words,  $y_n$  has a timeline with intervals of 2 seconds if  $x_n$  has a timeline with intervals of 1 second; the clock of the process is twice as slow. The process above is called *downsampling* by 2, and while simple, it has a number properties we will study in detail. For example, it is irreversible; once we remove samples, we cannot go back from  $y_n$  to  $x_n$ . It is also shift varying, requiring more complicated analysis.

The dual operation to downsampling is *upsampling*. For example, upsampling a sequence  $x_n$  by 2 results in a new sequence  $y_n$  by inserting zeros between every two samples as

$$y_n = \begin{cases} x_{n/2}, & n \text{ even;} \\ 0, & n \text{ odd,} \end{cases} \rightarrow [\dots x_{-1} \ 0 \ \boxed{x_0} \ 0 \ x_1 \ 0 \ \dots]^T. \quad (2.179)$$

The index of  $y_n$  corresponds to a time that is half of that for  $x_n$ . For example, if  $x_n$  has an interval of 1 second between samples, then  $y_n$  has intervals of 0.5 seconds; the clock of the process is twice as fast.

What we just saw for rate changes by 2 can be done for any integer as well as rational rate changes (the latter ones by combining upsampling by  $N$  and downsampling by  $M$ ). In addition, to smooth the sequence before dropping samples,



downsampling is preceded by lowpass filtering, while to fill in the zeros, upsampling is followed by filtering, thus combining filtering with sampling rate changes. The multirate operations are used in any number of today's physical systems, from MP3 players, to JPEG, MPEG, to name a few.

The purpose of this section is to study these various multirate operations and their consequences on the resulting sequences and their spectra. These operations turn multirate systems into *linear periodically shift-varying (LPSV)* systems, represented by block-Toeplitz instead of Toeplitz matrices. For example, in Section 2.3.1, we encountered a block-averaging operator, (2.48), which is linear but not shift invariant; it is, however, LPSV. If the period is an integer  $N$ , then the system is really the superposition of  $N$  LSI systems, and this is used in their analysis. While an LSI system is specified by its impulse response corresponding to a Kronecker delta sequence at time 0, an LPSV system is specified by  $N$  impulse responses corresponding to Kronecker delta sequences at times  $n = 0, 1, \dots, N - 1$ .

While the LPSV nature of multirate systems does complicate analysis, we use a relatively simple and powerful tool called polyphase analysis to mediate the problem and reduce it to the study of LSI systems. The outline of the section follows naturally the above introduction, moving from down- and upsampling together with filtering to polyphase analysis. Multirate processing, while not standard signal processing material, is central to filter bank and wavelet constructions. We summarize the main operations in multirate processing in Table 2.12, in *Chapter at a Glance*.

### 2.7.1 Downsampling

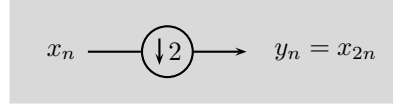
**Downsampling by 2** Downsampling<sup>56</sup> by 2, as introduced in (2.178) and shown in Figure 2.17(a), is clearly not shift invariant. If the input is  $x_n = \delta_n$ , the output is  $y_n = \delta_n$ ; if the input is  $x_n = \delta_{n-1}$ , the output is zero. However, if the input is  $x_n = \delta_{n-2k}$ , the output is  $y_n = \delta_{n-k}$ ; this means the downsampler is an LPSV system, making all multirate systems involving downsampling LPSV. It is instructive to look at (2.178) in matrix notation:

$$\begin{bmatrix} \vdots \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ y_2 \\ y_3 \\ \vdots \end{bmatrix} = \underbrace{\begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & \boxed{1} & 0 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \dots \\ \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{D_2} \begin{bmatrix} \vdots \\ x_{-2} \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ x_{-2} \\ \boxed{x_0} \\ x_2 \\ x_4 \\ x_6 \\ \vdots \end{bmatrix}, \quad (2.180a)$$

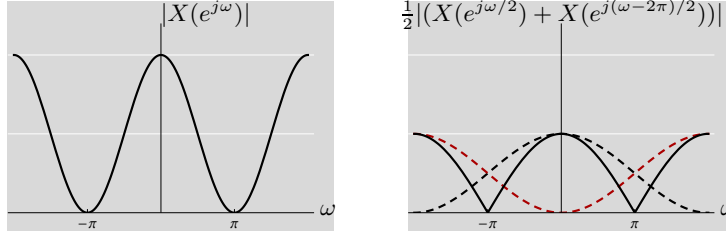
$$y = D_2 x, \quad (2.180b)$$

where  $D_2$  stands for the operator describing downsampling by 2. Inspection of  $D_2$  shows that it is similar to an identity matrix, but with the odd rows taken out.

<sup>56</sup>Downsampling is often referred to as subsampling and decimation as well.



(a) Block diagram.



(b) Magnitude response of a sequence and its (c) downsampled version.

**Figure 2.17:** Downsampling by 2.

Intuitively, it is a rectangular operator, with the output space being a subspace of the input space. It is a linear operator and has no inverse.

To find the  $z$ -transform  $Y(z)$ , the downsampled sequence may be seen as a sequence  $x_{0,n}$  having the even samples of  $x_n$  with the odd ones set to zero, followed by contraction of the sequence by removing those zeros. Thus,

$$x_{0,n} = \begin{cases} x_n, & n \text{ even;} \\ 0, & n \text{ odd.} \end{cases}$$

Its  $z$ -transform is

$$\begin{aligned} X_0(z) &= \frac{1}{2} [X(z) + X(-z)] \\ &= \frac{1}{2} [(\cdots + x_0 + x_1 z^{-1} + x_2 z^{-2} + \cdots) + (\cdots x_0 - x_1 z^{-1} + x_2 z^{-2} + \cdots)] \\ &= (\cdots + x_{-2} z^2 + x_0 + x_2 z^{-2} + \cdots) = \sum_{n \in \mathbb{Z}} x_{2n} z^{-2n}, \end{aligned}$$

canceling the odd powers of  $z$  and keeping the even ones. We now get  $Y(z)$  by contracting  $X_0(z)$  as:

$$Y(z) = \sum_{n \in \mathbb{Z}} x_{2n} z^{-n} = X_0(z^{1/2}) = \frac{1}{2} [X(z^{1/2}) + X(-z^{1/2})]. \quad (2.181)$$

To find its DTFT, we simply evaluate  $Y(z)$  at  $z = e^{j\omega}$ :

$$Y(e^{j\omega}) = \frac{1}{2} [X(e^{j\omega/2}) + X(e^{j(\omega-2\pi)/2})], \quad (2.182)$$

where  $-e^{j\omega/2}$  can be written as  $e^{j(\omega-2\pi)/2}$  since  $e^{-j\pi} = -1$ . With the help of Figure 2.17, we now analyze this formula.  $X(e^{j\omega/2})$  is a stretched version of  $X(e^{j\omega})$

(by a factor of 2) and is  $4\pi$ -periodic (since downsampling contracts time, it is natural that frequency expands accordingly). This is shown as the solid line in Figure 2.17(c).  $X(e^{j(\omega-2\pi)/2})$  is not only a stretched version of  $X(e^{j\omega})$ , but also shifted by  $2\pi$ , shown as the dashed line in Figure 2.17(c). The sum is again  $2\pi$ -periodic, since  $Y(e^{j\omega}) = Y(e^{j(\omega-2k\pi)})$ . Both stretching and shifting create new frequencies, an unusual occurrence, since in LSI processing, no new frequencies ever appear. The shifted version  $X(e^{j(\omega-2\pi)/2})$  is called the *aliased* version of the original (a ghost image).

**EXAMPLE 2.24 (DOWNSAMPLING OF SEQUENCES)** Consider first a standard example illustrating the effect of downsampling:  $x_n = (-1)^n = \cos \pi n$ , the highest-frequency discrete sequence. Its downsampled version is  $y_n = x_{2n} = \cos 2\pi n = 1$ , the lowest-frequency discrete sequence (a constant):

$$\left[ \dots \quad 1 \quad -1 \quad \boxed{1} \quad -1 \quad 1 \quad \dots \right]^T \xrightarrow{2\downarrow} \left[ \dots \quad 1 \quad 1 \quad \boxed{1} \quad 1 \quad 1 \quad \dots \right]^T$$

changing the nature of the sequence.

Consider now the right-sided geometric series sequence,  $x_n = \alpha^n u_n$ , from (2.124a), with the  $z$ -transform (from Table 2.6)

$$X(z) = \frac{1}{1 - \alpha z^{-1}}.$$

Its downsampled version is

$$y_n = x_{2n} = \alpha^{2n} u_{2n} \stackrel{(a)}{=} \beta^n u_n,$$

where (a) follows from  $u_{2n} = u_n$  and  $\beta = \alpha^2$ , with the  $z$ -transform (from Table 2.6)

$$Y(z) = \frac{1}{1 - \alpha^2 z^{-1}},$$

which we could have also obtained using the expression for downsampling (2.181)

$$\frac{1}{2} \left( \frac{1}{1 - \alpha z^{1/2}} + \frac{1}{1 + \alpha z^{-1/2}} \right) = \frac{1}{1 - \alpha^2 z^{-1}}.$$

The downsampled sequence is again exponential, but decays faster.

**Downsampling by  $N$**  We can generalize the discussion of downsampling by 2 to any integer  $N$ . The downsampled-by- $N$  sequence  $y_n$  and its  $z$ -transform pair are

$$y_n = x_{Nn} \xleftrightarrow{\text{ZT}} Y(z) = \frac{1}{N} \sum_{k=0}^{N-1} X(W_N^k z^{1/N}). \quad (2.183)$$

The corresponding DTFT pair on the unit circle is

$$y_n = x_{Nn} \xleftrightarrow{\text{DTFT}} Y(e^{j\omega}) = \frac{1}{N} \sum_{k=0}^{N-1} X(e^{j(\omega-2\pi k)/N}), \quad (2.184)$$

using

$$W_N^k z^{1/N} \Big|_{z=e^{j\omega}} = e^{-j(2\pi/N)k} e^{j\omega/N} = e^{j(\omega-2\pi k)/N}.$$

We have already seen these expressions in Section 2.4.3 and Table 2.4 as scaling in time. The proof is an extension of the  $N = 2$  case, and we leave it as Exercise 2.20.

### 2.7.2 Upsampling

**Upsampling by 2** Upsampling by 2 as introduced in (2.179) and shown in Figure 2.18(a) stretches time by a factor of 2. In matrix notation, similarly to (2.180a):

$$\begin{bmatrix} \vdots \\ y_{-2} \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ \vdots \end{bmatrix} = \underbrace{\begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & 1 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & 0 & \dots \\ \dots & 0 & \boxed{1} & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 1 & 0 & \dots \\ \dots & 0 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & 1 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{U_2} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ x_2 \\ x_3 \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ x_{-1} \\ 0 \\ \boxed{x_0} \\ 0 \\ x_1 \\ 0 \\ x_2 \\ \vdots \end{bmatrix}, \quad (2.185a)$$

$$y = U_2 x, \quad (2.185b)$$

where  $U_2$  stands for the upsampling-by-2 operator. The matrix  $U_2$  looks like an identity matrix with rows of zeros in between every two rows. Another way to look at it is as an identity matrix with every other column removed.

In the  $z$ -transform domain, the expression for upsampling by 2 is

$$Y(z) = \sum_{n \in \mathbb{Z}} y_n z^{-n} = \sum_{n \in \mathbb{Z}} x_n z^{-2n} = X(z^2), \quad (2.186)$$

since all odd terms of  $y_n$  are zero, while the even ones are  $y_{2n} = x_n$  following (2.179). In frequency domain,

$$Y(e^{j\omega}) = X(e^{j2\omega}), \quad (2.187)$$

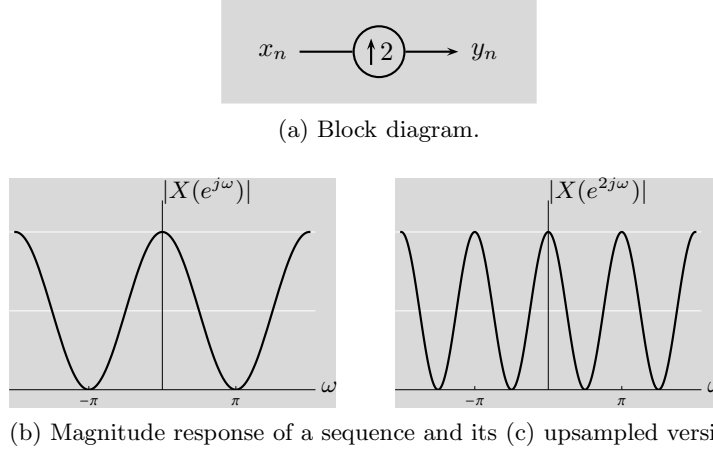
a contraction by a factor of 2 as shown in Figure 2.18(b) and (c).

**EXAMPLE 2.25** Take the constant sequence  $x_n = 1$ . Its upsampled version is

$y_n = [\dots \ 1 \ 0 \ \boxed{1} \ 0 \ 1 \ \dots]^T$ , and can be written as

$$y_n = \frac{1}{2} [1 + (-1)^n] = \frac{1}{2} (\cos 2\pi n + \cos \pi n),$$

indicating that it contains both the original frequency (constant, DC value at the origin) and a new high frequency (at  $\omega = \pi$ , since  $(-1)^n = e^{j\pi n} = \cos \pi n$ ).

**Figure 2.18:** Upsampling by 2.

**Upsampling by  $N$**  We can generalize the discussion of upsampling by 2 to any integer  $N$ . A sequence upsampled by  $N$  and its  $z$ -transform pair are given by

$$y_n = \begin{cases} x_{n/N}, & n = \ell N; \\ 0, & \text{otherwise,} \end{cases} \quad \xleftrightarrow{\text{ZT}} \quad Y(z) = X(z^N). \quad (2.188)$$

The corresponding DTFT pair on the unit circle is

$$y_n = \begin{cases} x_{n/N}, & n = \ell N; \\ 0, & \text{otherwise,} \end{cases} \quad \xleftrightarrow{\text{DTFT}} \quad Y(e^{j\omega}) = X(e^{jN\omega}). \quad (2.189)$$

### 2.7.3 Downsampling and Upsampling

The downsampling and upsampling operators are transposes of each other; since they have real entries, the upsampling operator is the adjoint of the downsampling operator,

$$U_2 = D_2^T = D_2^*. \quad (2.190)$$

**Upsampling Followed by Downsampling** What about combinations of upsampling and downsampling? Clearly, upsampling by 2 followed by downsampling by 2 results in the identity

$$D_2 U_2 = I, \quad (2.191)$$

since the zeros added by upsampling are at odd-indexed locations and are subsequently eliminated by downsampling. Similarly,

$$U_N D_N = I.$$

**Downsampling Followed by Upsampling** The operations in the reverse order is more interesting; downsampling by 2 followed by upsampling by 2 results in a sequence where all odd-indexed samples have been replaced by zeros, or

$$\left[ \dots \ x_{-1} \ x_0 \ x_1 \ x_2 \ \dots \right]^T \xrightarrow{2\downarrow, 2\uparrow} \left[ \dots \ 0 \ x_0 \ 0 \ x_2 \ \dots \right]^T$$

This operator,

$$P = U_2 D_2, \quad (2.192)$$

is an orthogonal projection operator onto the subspace of all even-indexed samples. To verify this, we check idempotency (for the operator to be a projection, see Definition 1.27),

$$P^2 = (U_2 D_2)(U_2 D_2) = U_2 (D_2 U_2) D_2 = U_2 D_2 = P,$$

using (2.191), as well as self-adjointness (for the operator to be an orthogonal projection, see Definition 1.27),

$$P^* = (U_2 D_2)^* = D_2^* U_2^* = U_2 D_2 = P,$$

using (2.190). Similarly,

$$D_N U_N = P_N,$$

where  $P_N$  is an orthogonal projection operator.

Applying this projection operator to a sequence  $x_n$ , we get the expressions in the DTFT and  $z$ -transform domain as:

$$P x_n \begin{array}{c} \xleftrightarrow{\text{DTFT}} \\ \xleftrightarrow{\text{ZT}} \end{array} \begin{array}{c} \frac{1}{2}(X(e^{j\omega}) + X(e^{j(\omega+\pi)})) \\ \frac{1}{2}(X(z) + X(-z)) \end{array} \quad (2.193)$$

## 2.7.4 Downsampling, Upsampling and Filtering

During both downsampling and upsampling processes, new frequencies appear, and it is often desirable to filter them out. While lowpass filtering is usually applied before downsampling to avoid aliasing (confusion of frequencies due to overlapping spectra), filtering is usually applied to fill in the zeros between the samples of the original sequence to smooth the output sequence. We now consider these two cases in more detail.

**Filtering Followed by Downsampling** Consider filtering followed by downsampling by 2, as illustrated in Figure 2.19. For simplicity, we look at a causal, length- $L$  FIR filter  $\tilde{g}$ ,<sup>57</sup> with operator  $\tilde{G}$  as in (1.228) (1.229), or (2.63). Then, convolution with

<sup>57</sup>From this point on, we use  $\tilde{g}$  to denote a filter when followed by a downsampler; similarly, we use  $g$  to denote a filter when preceded by an upsampler. This is done because in Part II of the book, this will be standard notation.

## 2.7. Multirate Sequences and Systems

255

the filter  $\tilde{g}_n$  followed by downsampling by 2 can be written as  $y = D_2 \tilde{G} x$ ,

$$\begin{bmatrix} \vdots \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ y_2 \\ \vdots \end{bmatrix} = \underbrace{\begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & \tilde{g}_1 & \tilde{g}_0 & 0 & 0 & 0 & 0 & \dots \\ \dots & \tilde{g}_3 & \tilde{g}_2 & \tilde{g}_1 & \boxed{\tilde{g}_0} & 0 & 0 & \dots \\ \dots & 0 & 0 & \tilde{g}_3 & \tilde{g}_2 & \tilde{g}_1 & \tilde{g}_0 & \dots \\ \dots & 0 & 0 & 0 & 0 & \tilde{g}_3 & \tilde{g}_2 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{D_2 \tilde{G}} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ x_2 \\ \vdots \end{bmatrix} = D_2 \tilde{G} x; \quad (2.194)$$

the above is nothing else but the convolution operator  $\tilde{G}$  with the odd rows removed. From this matrix-vector product, we can also write the filtered and downsampled sequence using inner products using (2.61a) as

$$y_n = \sum_{k=0}^{L-1} \tilde{g}_k x_{2n-k} = \langle x_{2n-k}, \tilde{g}_k \rangle_k = \sum_{k=0}^{L-1} x_k \tilde{g}_{2n-k} = \langle \tilde{g}_{2n-k}, x_k \rangle_k. \quad (2.195)$$

In the  $z$ -transform domain, apply (2.181) to  $\tilde{G}(z)X(z)$ ,

$$Y(z) = \frac{1}{2} \left[ \tilde{G}(z^{1/2})X(z^{1/2}) + \tilde{G}(-z^{1/2})X(-z^{1/2}) \right], \quad (2.196a)$$

and, on the unit circle for the DTFT,

$$Y(e^{j\omega}) = \frac{1}{2} \left[ \tilde{G}(e^{j\omega/2})X(e^{j\omega/2}) + \tilde{G}(e^{j(\omega-2\pi)/2})X(e^{j(\omega-2\pi)/2}) \right]. \quad (2.196b)$$

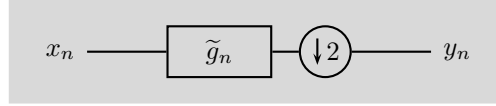
Figure 2.19 shows an input spectrum and its downsampled version. Figure 2.17(c) shows the spectrum without filtering, while Figure 2.19(c) shows the spectrum with filtering. Thus, when no filtering is used, aliasing perturbs the spectrum. When ideal lowpass filtering is used (Figure 2.19(b)), the spectrum from  $-\pi/2$  to  $\pi/2$  is conserved, the rest is put to zero so that no aliasing occurs, and the central lowpass part of the spectrum is conserved in the downsampled version (Figure 2.19(c)).

**EXAMPLE 2.26 (FILTERING FOLLOWED BY DOWNSAMPLING)** Consider the 2-point averaging filter  $\tilde{g}_n = \frac{1}{2}(\delta_n + \delta_{n-1})$ , whose output, downsampled by 2,  $y = D_2 \tilde{G} x$ , is

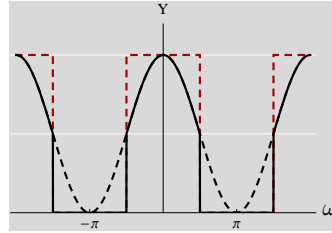
$$y_n = \frac{1}{2}(x_{2n} + x_{2n-1}).$$

Because of filtering, all input samples influence the output, as opposed to downsampling without filtering, where the odd-indexed samples had no impact.

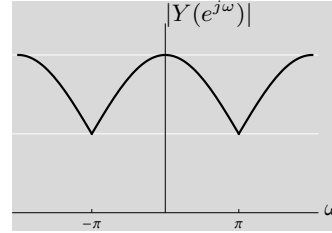
**Upsampling Followed by Filtering** Consider now upsampling followed by filtering, as shown in Figure 2.20. Using the matrix-vector notation, we can write the output



(a) Block diagram.



(b) Filtered sequence.



(c) Downsampled sequence.

**Figure 2.19:** Filtering and downsampling.

as the product  $GU_2$ , where  $G$  is a banded Toeplitz operator just like  $\tilde{G}$ :

$$\begin{bmatrix} \vdots \\ y_{-1} \\ \boxed{y_0} \\ y_1 \\ y_2 \\ \vdots \end{bmatrix} = \underbrace{\begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & g_2 & g_0 & 0 & 0 & \dots \\ \dots & g_3 & g_1 & 0 & 0 & \dots \\ \dots & 0 & g_2 & \boxed{g_0} & 0 & \dots \\ \dots & 0 & g_3 & g_1 & 0 & \dots \\ \dots & 0 & 0 & g_2 & g_0 & \dots \\ \dots & 0 & 0 & g_3 & g_1 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{GU_2} \begin{bmatrix} \vdots \\ x_{-1} \\ \boxed{x_0} \\ x_1 \\ x_2 \\ \vdots \end{bmatrix} = GU_2 x; \quad (2.197)$$

again, this is nothing else but the convolution operator  $G$  with the odd columns removed. Using inner products, we can express  $y_n$  as

$$y_n = \sum_k g_{n-2k} x_k = \langle x_k, g_{n-2k} \rangle_k. \quad (2.198)$$

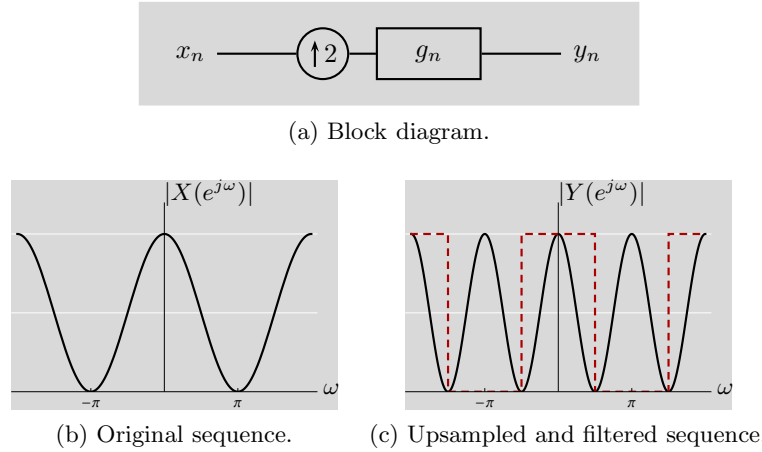
Another way to look at (2.197)–(2.198) is to see that each input sample  $x_k$  generates an impulse response  $g_n$  delayed by  $2k$  samples and weighted by  $x_k$ . In the  $z$ -transform and DTFT domains, the output of upsampling followed by filtering is

$$Y(z) = G(z)X(z^2), \quad (2.199a)$$

$$Y(e^{j\omega}) = G(e^{j\omega})X(e^{j2\omega}). \quad (2.199b)$$

Figure 2.20 shows a spectrum, its upsampled version, and finally, its ideally filtered version. We see that the ghost spectrum at  $2\pi$  is removed, and only the base spectrum around the origin remains.



**Figure 2.20:** Upsampling and filtering.

**EXAMPLE 2.27 (UPSAMPLING AND FILTERING)** Consider the piecewise constant filter  $g_n$  with impulse response  $g_n = \delta_n + \delta_{n-1}$ . The sequence  $x_n$ , upsampled by 2 and filtered with  $g_n$  leads to

$$y_n = [\dots \ x_{-1} \ x_{-1} \ x_0 \ x_0 \ x_1 \ x_1 \ \dots]^T, \quad (2.200)$$

a staircase sequence, with stairs of height  $x_n$  and length 2. A smoother interpolation is obtained with a linear interpolator:

$$g_n = \frac{1}{2}\delta_{n-1} + \delta_n + \frac{1}{2}\delta_{n+1}.$$

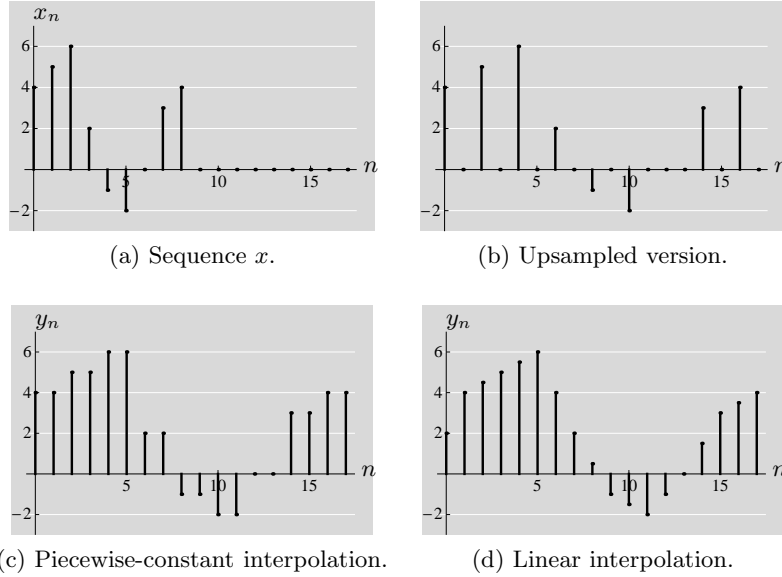
From (2.197) or (2.198), the even-indexed outputs are equal to input samples (at half the index), while odd-indexed outputs are averages of two input samples,

$$y_n = \begin{cases} x_{n/2}, & n \text{ even;} \\ \frac{1}{2}(x_{(n+1)/2} + x_{(n-1)/2}), & n \text{ odd.} \end{cases} \quad (2.201)$$

$$y = [\dots \ x_{-1} \ \frac{1}{2}(x_{-1} + x_0) \ x_0 \ \frac{1}{2}(x_0 + x_1) \ x_1 \ \dots]^T.$$

Compare (2.201) with (2.200) to see why (2.201) is a smoother interpolation, and see Figure 2.21 for an example.

**Downsampling, Upsampling and Filtering** Earlier, we noted the duality of downsampling and upsampling, made explicit in the adjoint relation (2.190). What happens when filtering is involved, when is  $D_2\tilde{G}$  the adjoint of  $GU_2$ ? By inspection,

**Figure 2.21:** Upsampling and filtering.

this holds when  $\tilde{g}_n^* = g_{-n}$ , since then<sup>58</sup>

$$\begin{aligned}
 (D_2 \tilde{G})^* &= \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \cdots & \tilde{g}_{-2} & \tilde{g}_0 & \tilde{g}_2 & \tilde{g}_4 & \tilde{g}_6 & \cdots \\ \cdots & \tilde{g}_{-3} & \tilde{g}_{-1} & \tilde{g}_1 & \tilde{g}_3 & \tilde{g}_5 & \cdots \\ \cdots & \tilde{g}_{-4} & \tilde{g}_{-2} & \boxed{\tilde{g}_0} & \tilde{g}_2 & \tilde{g}_4 & \cdots \\ \cdots & \tilde{g}_{-5} & \tilde{g}_{-3} & \tilde{g}_{-1} & \tilde{g}_1 & \tilde{g}_3 & \cdots \\ \cdots & \tilde{g}_{-6} & \tilde{g}_{-4} & \tilde{g}_{-2} & \tilde{g}_0 & \tilde{g}_2 & \cdots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \\
 &= \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \cdots & g_2 & g_0 & g_{-2} & g_{-4} & g_{-6} & \cdots \\ \cdots & g_3 & g_1 & g_{-1} & g_{-3} & g_{-5} & \cdots \\ \cdots & g_4 & g_2 & \boxed{g_0} & g_{-2} & g_{-4} & \cdots \\ \cdots & g_5 & g_3 & g_1 & g_{-1} & g_{-3} & \cdots \\ \cdots & g_6 & g_4 & g_2 & g_0 & g_{-2} & \cdots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} = GU_2. \quad (2.202)
 \end{aligned}$$

We could also show the above by using the definition of the adjoint operator in (1.44). That is, we want to find a  $\tilde{G}$  so that  $D_2 \tilde{G}$  and  $GU_2$  are adjoints of each

<sup>58</sup>Note that unlike in (2.194) and (2.197), here we do not assume causal filters.

other,

$$\langle D_2 \tilde{G} x, y \rangle = \langle x, GU_2 y \rangle. \quad (2.203)$$

We thus write

$$\langle D_2 \tilde{G} x, y \rangle \stackrel{(a)}{=} \langle \tilde{G} x, D_2^* y \rangle \stackrel{(b)}{=} \langle \tilde{G} x, U_2 y \rangle \stackrel{(c)}{=} \langle x, \tilde{G}^* U_2 y \rangle,$$

where (a) and (c) follow from conjugate linearity of the inner product in the second argument; and (b) follows from (2.190). Then (2.203) will hold only if  $\tilde{G}^* = G$ , that is,  $\tilde{g}_n^* = g_{-n}$ .

The above *Hermitian transposition equals time-reversal* will appear prominently in our analysis of orthogonal filter banks in Chapter 7. In particular, we will prove that when the impulse response of the filter  $g_n$  is orthogonal to its even shifts as in (2.204), and  $\tilde{g}_n^* = g_{-n}$ , then the operation of filtering, downsampling by 2, upsampling by 2 and filtering as in (7.18) is an orthogonal projection onto the subspace spanned by  $g_n$  and its even shifts.

### 2.7.5 Multirate Identities

**Orthogonality of Filter's Impulse Response to its Even Shifts** Filters that have impulse responses orthogonal to their even shifts

$$\langle g_n, g_{n-2k} \rangle = \delta_k, \quad (2.204)$$

will play an important role in the analysis of filter banks. Geometrically, (2.204) means that the columns of  $GU_2$  in (2.197) are orthonormal to each other (similarly for the rows of  $D_2 \tilde{G}$ ), that is,

$$I = (GU_2)^*(GU_2) = U_2^* G^* GU_2 = D_2 G^* GU_2. \quad (2.205)$$

While we have written the above in its most general form using Hermitian transposition to allow for complex filters, most of the time, we will be dealing with real filters, and thus simple transposition.

We can see (2.204) as the deterministic autocorrelation of  $g$  downsampled by 2. Write the deterministic autocorrelation of  $g$  as in (2.16)

$$a_k = \langle g_n, g_{n-k} \rangle_n,$$

and note that it has a single nonzero even term,  $g_0 = 1$ ,

$$a_{2k} = \delta_k. \quad (2.206)$$

Assume now a real  $g$ , in the  $z$ -transform domain,  $A(z) = G(z)G(z^{-1})$  using (2.142). Keeping only the even terms can be accomplished by adding  $A(z)$  and  $A(-z)$  and dividing by 2. Therefore, (2.206) can be expressed as

$$A(z) + A(-z) = G(z)G(z^{-1}) + G(-z)G(-z^{-1}) = 2, \quad (2.207)$$

which on the unit circle leads to

$$|G(e^{j\omega})|^2 + |G(e^{j(\omega+\pi)})|^2 = 2. \quad (2.208)$$

This *quadrature mirror formula*, also called *power complementarity*, will be central in the design of orthonormal filter banks in Chapter 7. In the above we have assumed that  $g_n$  is real, and used both

$$G(z)G(z^{-1})|_{z=e^{j\omega}} = G(e^{j\omega})G(e^{-j\omega}) = G(e^{j\omega})G^*(e^{j\omega}) = |G(e^{j\omega})|^2,$$

as well as

$$G(-z)G(-z^{-1})|_{z=e^{j\omega}} = |G(e^{j(\omega+\pi)})|^2.$$

In summary, a real filter satisfying any of the conditions below is called *orthogonal*:

$$\begin{array}{ccc} \langle g_n, g_{n-2k} \rangle = \delta_k & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_2 G^T G U_2 = I \\ G(z)G(z^{-1}) + G(-z)G(-z^{-1}) = 2 \\ |G(e^{j\omega})|^2 + |G(e^{j(\omega+\pi)})|^2 = 2 \end{array} \end{array} \quad (2.209)$$

Compare this to the expression for the allpass filter in (2.114); that allpass impulse response  $h$  is orthogonal to all its shifts, at the expense of no frequency selectivity. On the other hand, here we have a basis containing only even shifts; however, some frequency selectivity exists ( $g$  is a halfband lowpass filter). These issues will be explored in more details in Chapter 7.

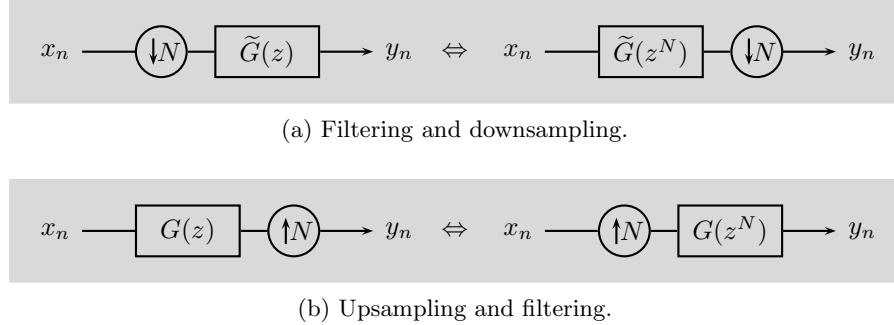
**Interchange of Multirate Operations and Filtering** The first of the two identities states that downsampling by 2 followed by filtering with  $\tilde{G}(z)$  is equivalent to filtering with  $\tilde{G}(z^2)$  followed by downsampling by 2, as shown in Figure 2.22(a).

The second identity states that filtering with  $G(z)$  followed by upsampling by 2 is equivalent to upsampling by 2 followed by filtering with  $G(z^2)$ , shown in Figure 2.22(b). The proof of these identities is left as Exercise 2.23, and both results generalize to sampling rate changes by  $N$ .

**Commutativity of Upsampling and Downsampling** Up/downsampling by the same integer do not commute, since, as we have seen,  $U_2$  followed by  $D_2$  is the identity, while  $D_2$  followed by  $U_2$  is a projection onto the subspace of even-indexed samples. Interestingly, upsampling by  $N$  and downsampling by  $M$  commute when  $N$  and  $M$  are relatively prime (that is, they have no common factor). The proof is the topic of Exercise 2.26, and a couple of illustrative examples are given in Exercise 2.27.

**EXAMPLE 2.28 (COMMUTATIVITY OF UPSAMPLING AND DOWNSAMPLING)** We look at upsampling by 2 and downsampling by 3. If we apply  $U_2$  to  $x_n$

$$U_2 x = [\dots \ x_0 \ 0 \ x_1 \ 0 \ x_2 \ 0 \ x_3 \ 0 \ \dots]^T,$$

**Figure 2.22:** Interchange of multirate operations and filtering.

followed by  $D_3$ , we get

$$D_3 U_2 x = [\dots \ x_{-3} \ 0 \ x_0 \ 0 \ x_3 \ 0 \ x_6 \ \dots]^T,$$

while, applying  $D_3$  first

$$D_3 x = [\dots \ x_{-3} \ x_0 \ x_3 \ x_6 \ \dots]^T,$$

followed by  $U_2$  leads to the same result,  $U_2 D_3 x = D_3 U_2 x$ .

### 2.7.6 Polyphase Representation

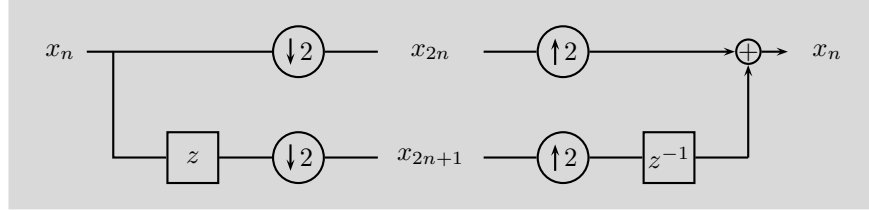
Multirate processing brings a major twist to signal processing: shift invariance is replaced by periodic shift variance, represented by block-Toeplitz matrices. This section examines a tool, called *polyphase representation*, that deals with such periodic shift variance. It is a key method to transform single-input single-output linear periodically shift varying systems into multiple-input multiple-output linear shift-invariant systems. For simplicity, we introduce all of the concepts for downsampling/upsampling by 2, and generalize to  $N$  only at the end of the section.

**Polyphase Representation of Sequences** A convenient way to express shift variance of period 2 is to split the input  $x_n$  into its even- and odd-indexed parts:

$$\begin{aligned} x_{0,n} &= x_{2n} & \xleftrightarrow{\text{ZT}} X_0(z) &= \sum_{n \in \mathbb{Z}} x_{2n} z^{-n}, \\ x_{1,n} &= x_{2n+1} & \xleftrightarrow{\text{ZT}} X_1(z) &= \sum_{n \in \mathbb{Z}} x_{2n+1} z^{-n}, \\ x_n &= \begin{cases} x_{0,n/2}, & n = 2k \\ x_{1,(n-1)/2}, & n = 2k+1 \end{cases} & \xleftrightarrow{\text{ZT}} X(z) &= X_0(z^2) + z^{-1} X_1(z^2). \end{aligned} \tag{2.210}$$

In the above,  $x_0$  is the even subsequence of  $x$  downsampled by 2 and  $x_1$  is the odd subsequence of  $x$  downsampled by 2:

$$\begin{aligned} x_0 &= [\dots \ x_{-2} \ \boxed{x_0} \ x_2 \ x_4 \ \dots]^T, \\ x_1 &= [\dots \ x_{-1} \ \boxed{x_1} \ x_3 \ x_5 \ \dots]^T. \end{aligned}$$

**Figure 2.23:** Forward and inverse polyphase transform.

This is illustrated in Figure 2.23: to get the even subsequence, we simply remove the odd samples from  $x$ , while to get the odd subsequence, we shift  $x$  by one to the left (advance by one represented by  $z$ ) and then remove the odd samples. To get the original sequence back, we revert the process: we upsample each subsequence by 2, shift the odd one by one to the right (delay by one represented by  $z^{-1}$ ), and sum up. This very simple transform that separates  $x$  into  $x_0$  and  $x_1$  is called the *forward polyphase transform*, with  $x_0$  and  $x_1$  the *polyphase components* of  $x$ . The *inverse polyphase transform* simply upsamples the two polyphase components and puts them back together by interleaving, as in (2.210).

For example, in polyphase transform domain, the effect of the operator  $Y = U_2 D_2 X$  is very simple:  $X_0(z)$  is kept while  $X_1(z)$  is zeroed out:

$$Y(z) = X_0(z^2).$$

Let us call  $a_{0,n}$  the deterministic autocorrelation sequence of the polyphase component  $x_{0,n} = x_{2n}$ , and, similarly, call  $a_{1,n}$  the deterministic autocorrelation sequence of the polyphase component  $x_{1,n} = x_{2n+1}$ . Their deterministic crosscorrelation will be  $c_{0,1,n}$ . Then, using the same polyphase tools, we can represent the deterministic autocorrelation (2.142) via its polyphase components (for simplicity, we show it for a real sequence  $x$  and in  $z$ -transform-domain):

$$\begin{aligned} A(z) &= X(z) X(z^{-1}) \\ &= (X_0(z^2) + z^{-1} X_1(z^2))(X_0(z^{-2}) + z X_1(z^{-2})) \\ &= (A_0(z^2) + A_1(z^2)) + (C_{1,0}(z^2) + z^2 C_{0,1}(z^2)), \end{aligned} \quad (2.211)$$

where the first and second terms are the first and second polyphase components of the deterministic autocorrelation, respectively, and satisfy:

$$A_0(z^2) + A_1(z^2) = A_0(z^{-2}) + A_1(z^{-2}), \quad (2.212a)$$

$$C_{1,0}(z^2) + z^2 C_{0,1}(z^2) = z^2 (C_{1,0}(z^{-2}) + z^{-2} C_{0,1}(z^{-2})). \quad (2.212b)$$

We can express the above in matrix form if we assume the  $x_n$  to be a vector sequence with its polyphase components as elements. Then, given  $x_n$ ,

$$x = \begin{bmatrix} x_0 & x_1 \end{bmatrix}^T, \quad (2.213)$$

## 2.7. Multirate Sequences and Systems

263

using (2.23), its deterministic autocorrelation is a matrix given by<sup>59</sup>

$$A_n = \begin{bmatrix} a_{0,n} & c_{0,1,n} \\ c_{0,1,-n}^* & a_{1,n} \end{bmatrix} = A_{-n}^*, \quad (2.214a)$$

$$A(e^{j\omega}) = \begin{bmatrix} A_0(e^{j\omega}) & C_{0,1}(e^{j\omega}) \\ C_{0,1}^*(e^{j\omega}) & A_1(e^{j\omega}) \end{bmatrix} = A^*(e^{j\omega}), \quad (2.214b)$$

$$A(z) = \begin{bmatrix} A_0(z) & C_{0,1}(z) \\ C_{0,1}^*(z^{-1}) & A_1(z) \end{bmatrix} = A_*(z^{-1}). \quad (2.214c)$$

**Polyphase Representation of Filtering** The next element we must examine is a filter. We decompose a filter exactly as we do a sequence in (2.210), into its even and odd subsequences as with

$$\begin{aligned} g_0 &= [\dots \ g_{-2} \ \boxed{g_0} \ g_2 \ g_4 \ \dots]^T, \\ g_1 &= [\dots \ g_{-1} \ \boxed{g_1} \ g_3 \ g_5 \ \dots]^T, \end{aligned}$$

leading to

$$g_{0,n} = g_{2n} \xleftrightarrow{\text{ZT}} G_0(z) = \sum_{n \in \mathbb{Z}} g_{2n} z^{-n}, \quad (2.215a)$$

$$g_{1,n} = g_{2n+1} \xleftrightarrow{\text{ZT}} G_1(z) = \sum_{n \in \mathbb{Z}} g_{2n+1} z^{-n}, \quad (2.215b)$$

$$G(z) = G_0(z^2) + z^{-1}G_1(z^2). \quad (2.215c)$$

This decomposition can be described via the polyphase representation of filtering (convolution), mapping the  $z$ -transform of a filter,  $G(z)$ , into its polyphase representation  $G_p(z)$ :

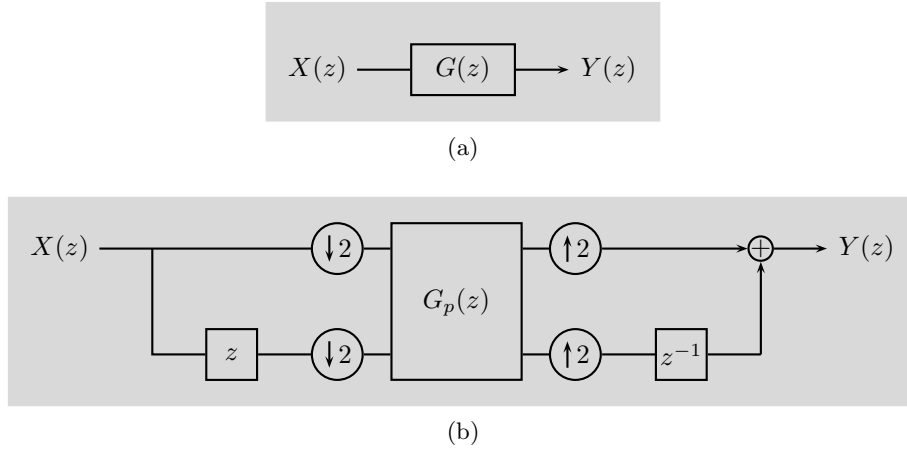
$$G_p(z) = \begin{bmatrix} G_0(z) & z^{-1}G_1(z) \\ G_1(z) & G_0(z) \end{bmatrix}, \quad (2.216)$$

depicted in Figure 2.24. The matrix  $G_p(z)$  is pseudocirculant (see Appendix 2.B.2). We can easily check that (2.216) is true since, according to Figure 2.24,

$$\begin{aligned} Y(z) &= \begin{bmatrix} 1 & z^{-1} \end{bmatrix} G_p(z^2) \begin{bmatrix} X_0(z^2) \\ X_1(z^2) \end{bmatrix} \\ &= \begin{bmatrix} 1 & z^{-1} \end{bmatrix} \begin{bmatrix} G_0(z^2)X_0(z^2) + z^{-2}G_1(z^2)X_1(z^2) \\ G_1(z^2)X_0(z^2) + G_0(z^2)X_1(z^2) \end{bmatrix} \\ &= G_0(z^2)X_0(z^2) + z^{-2}G_1(z^2)X_1(z^2) + z^{-1}(G_1(z^2)X_0(z^2) + G_0(z^2)X_1(z^2)) \\ &= [G_0(z^2) + z^{-1}G_1(z^2)][X_0(z^2) + z^{-1}X_1(z^2)] = G(z)X(z). \end{aligned}$$

We have seen in (2.202) that the adjoint of the filter operator  $G$  is  $\tilde{G} = G^*$ , when  $\tilde{g}_n = g_{-n}^*$ . Using (2.216), we find the polyphase representation of that adjoint

<sup>59</sup>See Footnote 52 on Page 236.



**Figure 2.24:** (a) Filtering and (b) its polyphase representation.

as (assuming real filter coefficients):

$$G_p^T(z^{-1}) = \begin{bmatrix} G_0(z^{-1}) & G_1(z^{-1}) \\ zG_1(z^{-1}) & G_0(z^{-1}) \end{bmatrix} = \begin{bmatrix} \tilde{G}_0(z) & z^{-1}\tilde{G}_1(z) \\ \tilde{G}_1(z) & \tilde{G}_0(z) \end{bmatrix} = \tilde{G}_p(z), \quad (2.217)$$

as expected.

**EXAMPLE 2.29 (POLYPHASE REPRESENTATION OF FILTERING)** Take  $G(z) = 1 + z^{-1}$ . Then, its polyphase representation (2.216) is

$$G_p(z) = \begin{bmatrix} 1 & z^{-1} \\ 1 & 1 \end{bmatrix}. \quad (2.218)$$

Its adjoint is

$$G_p^T(z^{-1}) = \begin{bmatrix} 1 & 1 \\ z & 1 \end{bmatrix} = \begin{bmatrix} \tilde{G}_0(z) & z^{-1}\tilde{G}_1(z) \\ \tilde{G}_1(z) & \tilde{G}_0(z) \end{bmatrix}, \quad (2.219)$$

yielding  $\tilde{G}(z) = 1 + z$ .

**Polyphase Representation of Upsampling Followed by Filtering** Instead of starting with the polyphase representation of filtering followed by downsampling, we first cover upsampling followed by filtering as it follows closely that of the polyphase representation of sequences we have just seen. Thus, upsampling by 2 followed by filtering with  $G(z)$  leads to an output  $Y(z)$ ,

$$Y(z) = G(z)X(z^2) = (G_0(z^2) + z^{-1}G_1(z^2))X(z^2) = Y_0(z^2) + z^{-1}Y_1(z^2),$$

where the polyphase components of  $Y(z)$  are  $Y_0(z) = G_0(z)X(z)$  and  $Y_1(z) = G_1(z)X(z)$ , as shown in Figure 2.25(a).



**Polyphase Representation of Filtering Followed by Downsampling** Given a polyphase representation of the input sequence  $x$  (2.210), we need to find the polyphase representation of the filter  $\tilde{g}$  so that the output after downsampling is correct. To do this, we call  $\tilde{g}_0$  and  $\tilde{g}_1$  the even and odd subsequences (polyphase components) of the filter. Then, observe that the convolution  $\tilde{g} * x$  can be written as:

$$\begin{aligned} (\tilde{g} * x)_n &= \sum_{k \in \mathbb{Z}} x_k \tilde{g}_{n-k} = \sum_{k \in \mathbb{Z}} x_{2k} \tilde{g}_{n-2k} + \sum_{k \in \mathbb{Z}} x_{2k+1} \tilde{g}_{n-2k-1} \\ &= (\tilde{g} * x_0)_n + (\tilde{g} * x_1)_n. \end{aligned}$$

Depending on  $n$ , the result of the above convolution will be:

$$\begin{aligned} n &= 2k & \tilde{g} * x &= \tilde{g}_0 * x_0 + \tilde{g}_1 * x_1, \\ n &= 2k+1 & \tilde{g} * x &= \tilde{g}_1 * x_0 + \tilde{g}_0 * x_1. \end{aligned}$$

However, since this convolution is followed by downsampling, the result of the convolution for  $n$  odd will disappear, leaving only

$$\tilde{g}_0 * x_0 + \tilde{g}_1 * x_1 \xleftrightarrow{\text{ZT}} \tilde{G}_0(z)X_0(z) + \tilde{G}_1(z)X_1(z). \quad (2.220)$$

We can use the above now to find the expression for the polyphase representation of a filter. Using (2.196a) to express the output of filtering followed by downsampling, the right side of (2.220) must equal:

$$\begin{aligned} \tilde{G}_0(z)X_0(z) + \tilde{G}_1(z)X_1(z) &= \frac{1}{2}[\tilde{G}(z^{1/2})X(z^{1/2}) + \tilde{G}(-z^{1/2})X(-z^{1/2})] \\ &\stackrel{(a)}{=} \frac{1}{2}[\underbrace{(\tilde{G}_0(z) + z^{k/2}\tilde{G}_1(z))}_{\tilde{G}(z^{1/2})} \underbrace{(X_0(z) + z^{-1/2}X_1(z))}_{X(z^{1/2})} + \\ &\quad \underbrace{(\tilde{G}_0(z) - z^{k/2}\tilde{G}_1(z))}_{\tilde{G}(-z^{1/2})} \underbrace{(X_0(z) - z^{-1/2}X_1(z))}_{X(-z^{1/2})}], \end{aligned}$$

where in (a) we substituted  $X(z)$  in its polyphase form using (2.210) and  $\tilde{G}(z)$  in a general polyphase form (we put in  $k$  as we do not know whether this polyphase form will involve a delay or an advance). Rearrange the right-hand side as

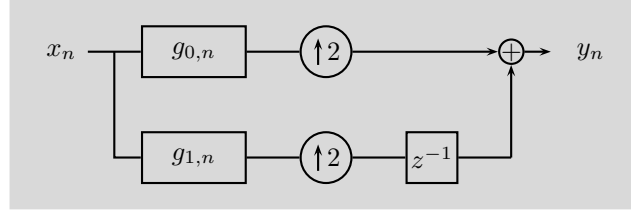
$$\begin{aligned} &\tilde{G}_0(z)X_0(z) + z^{k/2-1/2}\tilde{G}_1(z)X_1(z) + \\ &\frac{1}{2} \left( z^{-1/2}\tilde{G}_0(z)X_1(z) + z^{k/2}\tilde{G}_1(z)X_0(z) - z^{-1/2}\tilde{G}_0(z)X_1(z) - z^{k/2}\tilde{G}_1(z)X_0(z) \right). \end{aligned}$$

For this expression to equal the right-hand side of (2.220), the first summand must be equal to it and the second must be zero. This can be accomplished only if  $k = 1$  and the polyphase representation of the filter is

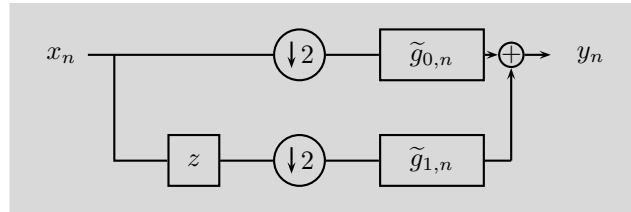
$$\tilde{g}_{0,n} = \tilde{g}_{2n} \xleftrightarrow{\text{ZT}} \tilde{G}_0(z) = \sum_{n \in \mathbb{Z}} \tilde{g}_{2n} z^{-n}, \quad (2.221a)$$

$$\tilde{g}_{1,n} = \tilde{g}_{2n-1} \xleftrightarrow{\text{ZT}} \tilde{G}_1(z) = \sum_{n \in \mathbb{Z}} \tilde{g}_{2n-1} z^{-n}, \quad (2.221b)$$

$$\tilde{G}(z) = \tilde{G}_0(z^2) + z\tilde{G}_1(z^2), \quad (2.221c)$$



(a) Upsampling and filtering from Figure 2.20(a).



(b) Filtering and downsampling from Figure 2.19(a).

**Figure 2.25:** Polyphase representation of multirate operations. Note that the definitions of polyphase components in (a) and (b) are different; see (2.215) and (2.221).

shown in Figure 2.25(b). Note  $z$  instead of  $z^{-1}$  in (2.210) and  $(2n-1)$  instead of  $(2n+1)$  in (2.210). The reason for  $(2n-1)$  in (2.221b) is because the 0th element of the polyphase component  $\tilde{g}_1$  is the sample at  $n = -1$ , that is,

$$\begin{aligned}\tilde{g}_0 &= \left[ \dots \quad \tilde{g}_{-2} \quad \boxed{\tilde{g}_0} \quad \tilde{g}_2 \quad \tilde{g}_4 \quad \dots \right]^T, \\ \tilde{g}_1 &= \left[ \dots \quad \tilde{g}_{-3} \quad \boxed{\tilde{g}_{-1}} \quad \tilde{g}_1 \quad \tilde{g}_3 \quad \dots \right]^T,\end{aligned}$$

and the polyphase component  $\tilde{g}_1$  must be advanced (shifted to the left) by one (multiplication by  $z$ ) after upsampling for proper reconstruction.

Note that the duality between (filtering, downsampling) and (upsampling, filtering) we have seen earlier, shows through the polyphase decomposition as well, (2.221c) and (2.215c). This duality, including the change from  $z^{-1}$  to  $z$  and from  $(2n-1)$  to  $(2n+1)$ , is related to the transposition and time reversal seen in (2.202).

**Polyphase Representation with Rate Changes by  $N$**  Generalization now follows naturally; the polyphase transform of size  $N$  decomposes a sequence into  $N$  phases

$$x_j = \left[ \dots \quad x_{N(n-1)+j} \quad x_{Nn+j} \quad x_{N(n+1)+j} \quad \dots \right]^T, \quad (2.222a)$$

## 2.7. Multirate Sequences and Systems

267

for  $j = 0, 1, \dots, N-1$ , leading to the expressions for a polyphase representation of a sequence, filtering followed by downsampling and upsampling followed by filtering:

$$x_{j,n} = x_{Nn+j} \xleftrightarrow{\text{ZT}} X_j(z) = \sum_{n \in \mathbb{Z}} x_{Nn+j} z^{-n}, \quad (2.222b)$$

$$X(z) = \sum_{j=0}^{N-1} z^{-j} X_j(z^N), \quad (2.222c)$$

$$\tilde{g}_{j,n} = \tilde{g}_{Nn-j} \xleftrightarrow{\text{ZT}} \tilde{G}_j(z) = \sum_{n \in \mathbb{Z}} \tilde{g}_{Nn-j} z^{-n}, \quad (2.222d)$$

$$\tilde{G}(z) = \sum_{j=0}^{N-1} z^j \tilde{G}_j(z^N), \quad (2.222e)$$

$$g_{j,n} = g_{Nn+j} \xleftrightarrow{\text{ZT}} G_j(z) = \sum_{n \in \mathbb{Z}} g_{Nn+j} z^{-n}, \quad (2.222f)$$

$$G(z) = \sum_{j=0}^{N-1} z^{-j} G_j(z^N). \quad (2.222g)$$

Note the difference between how polyphase components of  $\tilde{G}$  are defined compared to the polyphase components of  $G$ . Those for  $G$  are numbered forward modulo  $N$ , that is, the 0th polyphase component is the one at  $nN$ , the 1st is the one at  $nN+1$ , the 2nd at  $nN+2$ , and so on (same as for sequences). Those for  $\tilde{G}$ , on the other hand, are numbered in reverse modulo  $N$ , that is, the 0th polyphase component is the one at  $nN$ , but the 1st is the one at  $(Nn-1)$ , the 2nd at  $(Nn-2)$ , and so on, in reverse order from those for  $G$ . As illustration, for  $N=3$ , we give below the polyphase components of the input sequence, filtering followed by downsampling and upsampling followed by filtering, respectively:

$$x_0 = [\dots \ x_{-3} \ \boxed{x_0} \ x_3 \ x_6 \ \dots]^T,$$

$$x_1 = [\dots \ x_{-2} \ \boxed{x_1} \ x_4 \ x_7 \ \dots]^T,$$

$$x_2 = [\dots \ x_{-1} \ \boxed{x_2} \ x_5 \ x_8 \ \dots]^T,$$

$$\tilde{g}_0 = [\dots \ \tilde{g}_{-3} \ \boxed{\tilde{g}_0} \ \tilde{g}_3 \ \tilde{g}_6 \ \dots]^T,$$

$$\tilde{g}_1 = [\dots \ \tilde{g}_{-4} \ \boxed{\tilde{g}_{-1}} \ \tilde{g}_2 \ \tilde{g}_5 \ \dots]^T,$$

$$\tilde{g}_2 = [\dots \ \tilde{g}_{-5} \ \boxed{\tilde{g}_{-2}} \ \tilde{g}_1 \ \tilde{g}_4 \ \dots]^T,$$

$$g_0 = [\dots \ g_{-3} \ \boxed{g_0} \ g_3 \ g_6 \ \dots]^T,$$

$$g_1 = [\dots \ g_{-2} \ \boxed{g_1} \ g_4 \ g_7 \ \dots]^T,$$

$$g_2 = [\dots \ g_{-1} \ \boxed{g_2} \ g_5 \ g_8 \ \dots]^T.$$

Exercise 2.28 illustrates our discussion on polyphase transform.

## 2.8 Discrete Stochastic Processes and Systems

Many applications of signal processing involve resolving, reducing, or exploiting uncertainty. Resolving uncertainty includes identifying which out of a set of sequences was transmitted over a noisy channel; reducing uncertainty includes estimating parameters from noisy observations; and exploiting uncertainty includes cryptographic encoding in which the meanings of symbols are hidden from anyone lacking the key. Careful modeling of uncertainty is also exploited in compression when short descriptions are assigned to the most likely inputs. The issue of uncertainty arises in many other fields, where such processes are typically called *time series*, and the operations often include modeling of such time series (for example, modeling noise in a fluorescence microscope), estimating its parameters and predicting its future behavior (for example, the behavior of the stock market), among others.

As we have seen in Section 1.C, one of the tools for modeling uncertainty is probability theory, while deriving probabilistic models from the observed data is the task of statistics. In what follows, we discuss this stochastic framework within discrete-time signal processing, following the same structure of the chapter: We start with discrete stochastic processes (random processes, time series). We follow this with systems (almost exclusively LSI systems) and associated functions on discrete stochastic processes both in time domain (as averages in the form of means, variances and correlation functions), as well as in frequency domain (power spectral density). We finish with a brief analysis of multirate systems with stochastic behavior.

### 2.8.1 Processes

A discrete stochastic process is an infinite-length sequence whose every element is a random variable; in other words, it is a random process (see Section 1.C). For example, our temperature example from the introduction could be such a sequence; the temperature at noon in front of your house measured every day. If all random variables have the same distribution and are independent, the process is called *i.i.d.* (*independent and identically distributed*). We use the following functions defined on a stochastic process:<sup>60</sup>

mean	$\mu_{x,n}$	$E[x_n]$	
variance	$\text{var}(x_n)$	$E[(x_n - \mu_{x,n})^2]$	
standard deviation	$\sigma_{x,n}$	$\sqrt{\text{var}(x_n)}$	(2.223)
autocorrelation	$a_{x,k,n}$	$E[x_k x_{k-n}^*]$	
crosscorrelation	$c_{x,y,k,n}$	$E[x_k y_{k-n}^*]$	

Most of the time we will assume we are dealing with *wide-sense stationary* (WSS) processes, that is, those processes whose mean is constant and autocorrelation de-

<sup>60</sup> Although we have seen a version of these in Section 1.C, we repeat them here for completeness.

## 2.8. Discrete Stochastic Processes and Systems

269

pends only on  $n$ :

$$\mu_{x,n} = \mu_x; \quad a_{x,k,n} = a_{x,n}. \quad (2.224)$$

Often, this assumption is valid at least for a portion of time of a given sequence; for example, many biological processes are stationary over a period of milliseconds, while the noise in a communications channel might be stationary over a much longer period of time.

**White Noise** A very particular discrete stochastic process appearing widely in signal processing is the *white noise*<sup>61</sup> sequence  $x$ , whose mean is zero and autocorrelation  $a_n$  is  $a_n = \sigma_x^2 \delta_n$ , or,

$$\mu_{x,n} = 0; \quad \text{var}(x_n) = \sigma_x^2; \quad \sigma_{x,n} = \sigma_x; \quad a_{x,n} = \sigma_x^2 \delta_n. \quad (2.225)$$

If the underlying PDF is Gaussian,  $x$  is called *white Gaussian noise*.<sup>62</sup>

The white noise sequence is uncorrelated, but not always independent (in the case of the Gaussian PDF, it will automatically be independent as well). Often, the term *whitening*, or, *decorrelation* is used, meaning that a given sequence is made to have zero mean and single-term autocorrelation sequence, and is basically a diagonalization process for the covariance matrix.

## 2.8.2 Systems

We now assume that our input  $x$  is a WSS sequence, and the system is LSI described by its impulse response  $h$ , as in Section 2.3.3. What can we say about the output? It is given by (2.58), and we can compute the same functions on the output we computed on the input (mean, variance, standard deviation and autocorrelation). We start with the mean:

$$\begin{aligned} \mu_{y,n} &= E[y_n] \stackrel{(a)}{=} E\left[\sum_{k \in \mathbb{Z}} x_k h_{n-k}\right] \stackrel{(b)}{=} \sum_{k \in \mathbb{Z}} E[x_k] h_{n-k} \\ &\stackrel{(c)}{=} \mu_x \sum_{k \in \mathbb{Z}} h_{n-k} \stackrel{(d)}{=} \mu_x H(e^{j0}) = \mu_y, \end{aligned} \quad (2.226a)$$

where (a) follows from (2.58); (b) from the linearity of the expectation; (c) from  $x$  being WSS; and (d) from the frequency response of the LSI system (assuming it exists). We can thus see that the mean of the output is a constant, independent of  $n$ . The variance is

$$\text{var}(y_n) = a_{y,0} - (\mu_y)^2, \quad (2.226b)$$

<sup>61</sup>We will shortly see that the DTFT of the autocorrelation of white noise is a constant, mimicking the behavior of the spectrum of the white light; thus the term white noise.

<sup>62</sup>White Gaussian noise is often a mathematically usable model for many real-world processes, and is sometimes termed *additive white Gaussian noise* (AWGN). This is because in many models, it is added to the sequence of interest.

and the autocorrelation

$$\begin{aligned}
 a_{y,k,n} &= E[y_k y_{k-n}^*] \stackrel{(a)}{=} E\left[\sum_{m \in \mathbb{Z}} x_{k-m} h_m \sum_{\ell \in \mathbb{Z}} x_{k-n-\ell}^* h_\ell^*\right] \\
 &\stackrel{(b)}{=} \sum_{m \in \mathbb{Z}} \sum_{\ell \in \mathbb{Z}} h_m h_\ell^* E[x_{k-m} x_{k-n-\ell}^*] \stackrel{(c)}{=} \sum_{m \in \mathbb{Z}} \sum_{\ell \in \mathbb{Z}} h_m h_\ell^* a_{x,n-(m-\ell)} \\
 &\stackrel{(d)}{=} \sum_{p \in \mathbb{Z}} \left(\sum_{m \in \mathbb{Z}} h_m h_{m-p}^*\right) a_{x,n-p} \stackrel{(e)}{=} \sum_{p \in \mathbb{Z}} a_{h,p} a_{x,n-p} = a_{y,n}, \quad (2.226c)
 \end{aligned}$$

where (a) follows from the definition of convolution (2.58); (b) from linearity of expectation; (c) from the expression for the autocorrelation of the WSS  $x$ , (2.223); (d) from change of variable  $p = m - \ell$ ; and (e) from the definition of deterministic autocorrelation (2.16). From this, we see that if the input is WSS, the output is WSS as well (as the mean is a constant and the autocorrelation depends only on the difference  $n$ ). We also see that the autocorrelation of the output is the convolution of the stochastic autocorrelation of the input and the deterministic autocorrelation of the impulse response of the system.

Similarly, we compute the crosscorrelation between the input and the output:

$$\begin{aligned}
 c_{x,y,k,n} &= E[x_k y_{k-n}^*] \stackrel{(a)}{=} E\left[x_k \sum_{\ell \in \mathbb{Z}} h_\ell^* x_{k-n-\ell}^*\right] = E\left[\sum_{\ell \in \mathbb{Z}} h_\ell^* x_k x_{k-(n+\ell)}^*\right] \\
 &\stackrel{(b)}{=} \sum_{\ell \in \mathbb{Z}} h_\ell^* E[x_k x_{k-(n+\ell)}^*] \stackrel{(c)}{=} \sum_{\ell \in \mathbb{Z}} h_\ell^* a_{x,n+\ell}, \quad (2.226d)
 \end{aligned}$$

where (a) follows from (2.58); (b) from linearity of expectation; and (c) from the expression for the autocorrelation of the WSS  $x$ , (2.223). We will use the above expressions shortly to make some important observations in the Fourier domain.

**Autoregressive Moving-Average Model** Given the input  $x$  and an LSI system  $h$ , an *ARMA* model is used to analyze and possibly predict future values of the output, with only a finite number of parameters. It is given by:

$$y_n = \sum_{k=0}^M b_k x_{n-k} + \sum_{k=1}^N a_k y_{n-k}, \quad (2.227)$$

formally the same as the linear, constant-coefficient difference equation (2.53) (except for the sign in front of the second sum).

When all the  $b_k$ , except for  $b_0$ , are zero, the sequence is called an *autoregressive* (*AR*) process:

$$y_n = b_0 x_n + \sum_{k=1}^N a_k y_{n-k}.$$

## 2.8. Discrete Stochastic Processes and Systems

271

EXAMPLE 2.30 (FIRST-ORDER AR PROCESS) An AR-1 process is described via

$$y_n = x_n + a y_{n-1}, \quad (2.228)$$

that is, an AR process with  $N = 1$ ,  $b_0 = 1$  and  $a_1 = a$ . We normalize it to have unit norm, leading to:

$$h_n = \sqrt{1 - a^2} a^n u_n, \quad |a| < 1. \quad (2.229)$$

When, instead, all  $a_k$  are zero, the sequence is called a *moving average (MA)* process, as the output is a windowed, local, view of a number of inputs (see also Example 2.2):

$$y_n = \sum_{k=0}^M b_k x_{n-k}.$$

EXAMPLE 2.31 (FIRST-ORDER MA PROCESS) An MA-1 process is described via

$$y_n = b_0 x_n + b_1 x_{n-1}, \quad (2.230)$$

that is, an MA process with  $M = 1$ . We normalize it to have unit norm,  $b_0 = b_1 = 1/\sqrt{2}$ , leading to:

$$h_n = \frac{1}{\sqrt{2}}(\delta_n + \delta_{n-1}). \quad (2.231)$$

Thus, an ARMA process consists of two parts: the AR part and the MA part.

## 2.8.3 Discrete-Time Fourier Transform

As for deterministic sequences, we can use Fourier techniques to gain insight into the behavior of discrete stochastic processes and systems. While we cannot take a DTFT of a discrete stochastic process, as it is neither absolutely, nor square summable, we can make assessments based on averages (moments), such as, taking the DTFT of the autocorrelation. Let us assume a WSS  $x$  and find the DTFT of its autocorrelation (2.223) (which we assume to have sufficient decay so as to be absolutely summable):

$$A_x(e^{j\omega}) = \sum_{n \in \mathbb{Z}} a_{x,n} e^{-j\omega n} = \sum_{n \in \mathbb{Z}} \mathbb{E}[x_k x_{k-n}^*] e^{-j\omega n}. \quad (2.232)$$

This is called the *power spectral density*, the counterpart of the *energy spectral density* for deterministic sequences in (2.96). The power spectral density exists if and only if  $x$  is WSS, the result of the Wiener-Khinchin theorem. When  $x$  is real, the power spectral density is nonnegative, and thus admits a spectral factorization

$$A_x(e^{j\omega}) = U(e^{j\omega}) U^*(e^{j\omega}),$$

where  $U(e^{j\omega})$  is its (nonunique) spectral root. The integral of the power spectral density over the frequency range,

$$P_x = \frac{1}{2\pi} \int_{-\pi}^{\pi} A_x(e^{j\omega}) d\omega = a_{x,0} = E[|x_n|^2], \quad (2.233)$$

is the *power*, the counterpart of the *energy* for deterministic sequences in (2.98). The power spectral density measures the distribution of power over the frequency range. These four concepts, energy and energy spectral density for deterministic sequences, and power and power spectral density for WSS processes, are summarized in Table 2.8.

Deterministic sequences	discrete WSS processes
Energy spectral density $A(e^{j\omega}) =  X(e^{j\omega}) ^2$	Power spectral density $A(e^{j\omega}) = \sum_{n \in \mathbb{Z}} E[x_k x_{k-n}^*] e^{-j\omega n}$
Energy $E = \frac{1}{2\pi} \int_{-\pi}^{\pi} A(e^{j\omega}) d\omega$ $E = a_0 = \sum_{n \in \mathbb{Z}}  x_n ^2$	Power $P = \frac{1}{2\pi} \int_{-\pi}^{\pi} A(e^{j\omega}) d\omega$ $P = a_0 = E[ x_n ^2]$

**Table 2.8:** Energy concepts for the deterministic sequences and their counterparts, power concepts for discrete WSS processes.

EXAMPLE 2.32 (FIRST-ORDER AR PROCESS, EXAMPLE 2.30 CONT'D) Let us take a unit-norm version of an AR-1 process as in (2.229). Its power spectral density is

$$A_x(e^{j\omega}) = \frac{1 - a^2}{(1 - ae^{j\omega})(1 - ae^{-j\omega})} = \frac{1 - a^2}{|1 - ae^{-j\omega}|^2}, \quad |a| < 1. \quad (2.234)$$

This function is positive definite, that is,  $A_x(e^{j\omega}) > 0$ .

EXAMPLE 2.33 (FIRST-ORDER MA PROCESS, EXAMPLE 2.31 CONT'D) Let us take a unit-norm version of an MA-1 process as in (2.230). Its power spectral density is

$$A_x(e^{j\omega}) = \frac{1}{2}(1 + e^{j\omega})(1 + e^{-j\omega}) = 1 + \frac{1}{2}(e^{j\omega} + e^{-j\omega}) = 1 + \cos \omega. \quad (2.235)$$

It is positive semidefinite, that is,  $A_x(e^{j\omega}) \geq 0$ , being 0 for  $\omega = (2k + 1)\pi$ .

Given (2.226c), the autocorrelation of the output can be expressed as

$$A_y(e^{j\omega}) = A_h(e^{j\omega}) A_x(e^{j\omega}) = |H(e^{j\omega})|^2 A_x(e^{j\omega}), \quad (2.236)$$

where  $A_h(e^{j\omega}) = |H(e^{j\omega})|^2$  is the DTFT of the deterministic autocorrelation of  $h$ , according to Table 2.4. The quantity

$$P_y = E[y_n^2] = \frac{1}{2\pi} \int_{-\pi}^{\pi} A_y(e^{j\omega}) d\omega = \frac{1}{2\pi} \int_{-\pi}^{\pi} |H(e^{j\omega})|^2 A_x(e^{j\omega}) d\omega = a_y(0),$$



## 2.8. Discrete Stochastic Processes and Systems

273

is the *output power*. Similarly to (2.236), and from (2.226d), we can express the crosscorrelation between the input and the output as

$$C_{x,y}(e^{j\omega}) = H^*(e^{j\omega}) A_x(e^{j\omega}). \quad (2.237)$$

**White Noise** Using (2.225) and Table 2.4, we see that the power spectral density of white noise is a constant:

$$A(e^{j\omega}) = \sigma_x^2, \quad (2.238)$$

that is, its variance, or, power, is

$$a_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} \sigma_x^2 d\omega = \sigma_x^2.$$

**Power Spectral Density Estimation** Estimating power spectral density is of importance, as it describes a stochastic system. This is typically done by estimating its local behavior, requiring some form of a local Fourier transform. This local estimate is called a *periodogram*, and its definition and applications are given in Section 8.3.2, as they are implemented using filter banks.

**Orthogonality of WSS Processes** The concept of orthogonality of two deterministic sequences is defined as their inner product being zero. Just like for random vectors (see Appendix 1.C), we need an extension of this concept to handle stochastic processes. For simplicity, we consider only discrete WSS processes.

**DEFINITION 2.17** Two discrete WSS processes  $x_n$  and  $y_n$  are called orthogonal when

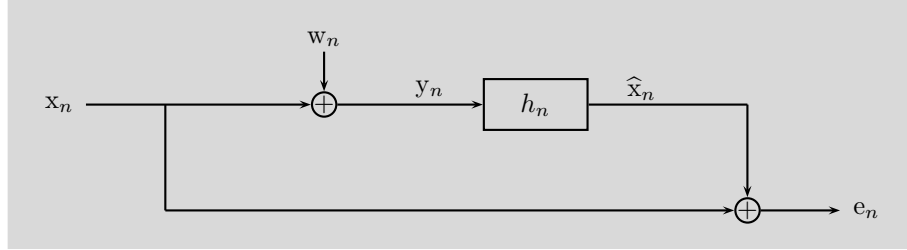
$$c_{x,y,n} = E[x_k y_{k-n}] = 0, \quad n \in \mathbb{Z}, \quad (2.239a)$$

$$C_{x,y}(e^{j\omega}) = 0, \quad \omega \in \mathbb{R}. \quad (2.239b)$$

Orthogonality properties for both deterministic sequences and discrete stochastic processes are summarized in Table 2.9.

	Deterministic sequence	WSS sequence
Time	$c_{x,y,n} = \sum_{k \in \mathbb{Z}} x_k y_{k-n}^* = \langle x_k, y_{k-n} \rangle_k = 0$	$c_{x,y,n} = E[x_k y_{k-n}] = 0$
Frequency	$C_{x,y}(e^{j\omega}) = X(e^{j\omega}) Y^*(e^{j\omega}) = 0$	$C_{x,y}(e^{j\omega}) = 0$

**Table 2.9:** Orthogonality for deterministic sequences and discrete stochastic processes.



**Figure 2.26:** Wiener filtering as the estimation of the original sequence  $x$  (corrupted by noise  $w$ ) by finding a filter with the impulse response  $h$  such that the estimate (filtered output)  $\hat{x}$  is a minimum MSE estimate of  $x$  by minimizing the squared error  $E[e_n^2]$ .

**EXAMPLE 2.34 (WIENER FILTERING)** Consider a zero-mean WSS sequence  $x_n$  with autocorrelation  $a_{x,n}$  and the corresponding DTFT  $A_x(e^{j\omega})$ . Consider now a version corrupted by noise as in Figure 2.26,

$$y_n = x_n + w_n, \quad (2.240a)$$

where  $w_n$  is additive, zero-mean, WSS noise with autocorrelation  $a_{w,n}$ , DTFT  $A_w(e^{j\omega})$ , and is independent of  $x_n$ . We want to estimate our original sequence  $x_n$  by finding a filter with the impulse response  $h_n$  such that the filtered output

$$\hat{x} = h * y = h * (x + w), \quad (2.240b)$$

is a minimum MSE estimate of  $x_n$ , by solving

$$\min_h E[e_n^2] = \min_h E[(x_n - \hat{x}_n)^2]. \quad (2.240c)$$

This minimization can be solved by writing  $e$  as a function of the impulse response of  $h$  and setting the derivatives with respect to  $h$  to zero (Exercise 2.7).

Instead, we are going to use a geometric approach. Because  $\hat{x} = h * y$ , the best estimate is the orthogonal projection of  $x$  onto the subspace  $S = \text{span}\{y_{n-k}\}_{k \in \mathbb{Z}}$ . The orthogonality principle states that the error  $e$  is orthogonal to the estimate  $\hat{x}$ . From Definition 2.17,

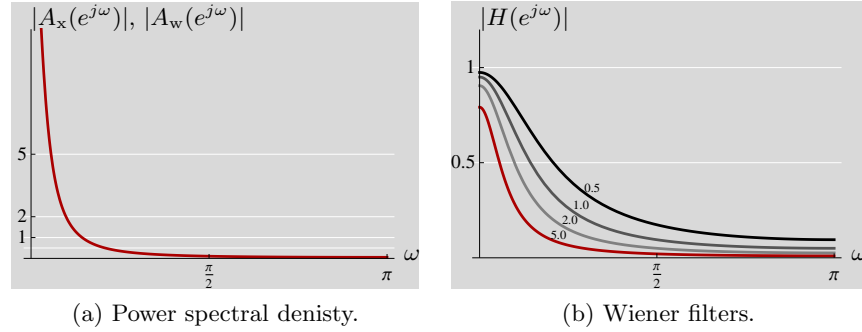
$$E[(x_n - \hat{x}_n) \hat{x}_{n-m}^*] = 0. \quad (2.241a)$$

Evaluating the two terms separately

$$c_{x, \hat{x}, m} = E[x_n \hat{x}_{n-m}^*] = E[x_n (h^* *_{n-m} (x^* + w^*))] \stackrel{(a)}{=} E[x_n (h^* *_{n-m} x^*)],$$

where (a) follows from the orthogonality of  $x$  and  $w$ . Rewriting this in the Fourier domain

$$C_{x, \hat{x}}(e^{j\omega}) = H^*(e^{j\omega}) A_x(e^{j\omega}). \quad (2.241b)$$



**Figure 2.27:** Wiener filtering. (a) Power spectral density  $A_x(e^{j\omega})$  of an AR-1 process with  $a = 0.9$  in (2.234) and noise levels of  $A_w(e^{j\omega}) = \sigma_w^2 \in \{0.5, 1, 2, 5\}$  (white grid lines). (b) Magnitude responses of the corresponding Wiener filters  $H(e^{j\omega})$  for noise levels of  $A_w(e^{j\omega}) = \sigma_w^2 \in \{0.5, 1, 2, 5\}$ .

Next, evaluate the second term in (2.241a),

$$c_{\hat{x}, \hat{x}, m} = E[\hat{x}_n \hat{x}_{n-m}^*] = E[h *_{n-m} (x + w) h^* *_{n-m} (x^* + w^*)].$$

Out of the four terms in the above equation, the two involving  $x$  and  $w$  are both zero by orthogonality, while the other two yield (in the Fourier domain),

$$C_{\hat{x}, \hat{x}}(e^{j\omega}) = |H(e^{j\omega})|^2 (A_x(e^{j\omega}) + A_w(e^{j\omega})). \quad (2.241c)$$

Substituting (2.241b) and (2.241c) into (2.241a) leads to

$$H^*(e^{j\omega}) A_x(e^{j\omega}) = |H(e^{j\omega})|^2 (A_x(e^{j\omega}) + A_w(e^{j\omega})),$$

and finally,

$$H(e^{j\omega}) = \frac{A_x(e^{j\omega})}{A_x(e^{j\omega}) + A_w(e^{j\omega})}. \quad (2.241d)$$

As a concrete example, choose  $x$  as an AR-1 process from Example 2.32 with  $A_x(e^{j\omega})$  as in (2.234). For  $w$ , choose white noise, (2.225), with variance  $\sigma_w^2$ , or,  $A_w(e^{j\omega}) = \sigma_w^2$ . Then, the Wiener filter is

$$H(e^{j\omega}) = \frac{1 - a^2}{1 - a^2 + \sigma_w^2 |1 - ae^{-j\omega}|^2}, \quad (2.242)$$

illustrated in Figure 2.27 for  $a = 0.9$  and various noise levels.

### 2.8.4 Multirate Sequences and Systems

An interesting question is what happens when a discrete stochastic process makes its way through a multirate system, with any combination of multirate operations we have seen so far. We will see that the notion of *periodic shift variance* for deterministic systems has its counterpart in the notion of *wide-sense cyclostationarity* for stochastic systems.

**DEFINITION 2.18** A discrete stochastic process  $x$  is called wide-sense cyclostationary of period  $N$  (WSCS <sub>$N$</sub> ) when the vector of its polyphase components is WSS.

Our temperature example comes in handy here as well; we take the temperature sequence  $x$  and decompose it into its polyphase components modulo 365. Then, every calendar day can follow its own statistical behavior. For example, the temperature at noon on January 14th in New York City will likely be low, while the same measurement taken on July 14th will likely be high. The notion of cyclostationarity for LPSV systems is very intuitive given their cyclic nature. We now discuss a few basic operations for illustration only; see *Further Reading* for pointers to literature.

In what follows, we will see that while downsampling affects the output power spectral density, it does not affect the WSS nature of the sequence. On the other hand, upsampling does affect the nature of the sequence and causes the output to become WSCS instead.

The mean and autocorrelation of a WSCS <sub>$N$</sub>  sequence  $x$  satisfy

$$\mu_{x,n+N} = E[x_{n+N}] = E[x_n] = \mu_{x,n}, \quad (2.243a)$$

$$a_{x,k+N,n} = E[x_{k+N} x_{k+N-n}^*] = E[x_k x_{k-n}^*] = a_{x,k,n}, \quad (2.243b)$$

and also

$$x \text{ is WSCS}_N \Rightarrow x \text{ is WSCS}_{kN}, \quad k \in \mathbb{Z}.$$

Beware that (2.243b) does not imply that  $a_{x,k,n}$  is periodic as we now illustrate:

**EXAMPLE 2.35 (GENERATIVE MODEL FOR A WSCS<sub>2</sub> SEQUENCE)** Given is a system as in Figure 2.28, with  $x$  an AWGN sequence, that is, it is  $\mathcal{N}(0, \sigma_x^2)$  (we can assume for simplicity it is  $\mathcal{N}(0, 1)$ ).<sup>63</sup> Then,  $y_n$  are independent random variables with the following distributions:

$$\begin{aligned} y_{2n} &\sim \mathcal{N}(0, \sigma_0^2), \\ y_{2n+1} &\sim \mathcal{N}(0, \sigma_1^2), \end{aligned}$$

and thus, from (2.225), their autocorrelations are:

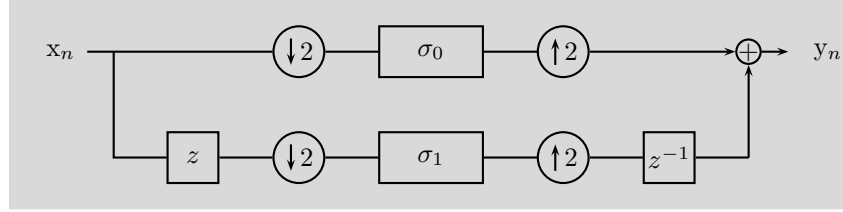
$$\begin{aligned} a_{y,2k,n} &= E[y_{2k} y_{2k-n}^*] = \sigma_0^2 \delta_n, \\ a_{y,2k+1,n} &= E[y_{2k+1} y_{2k+1-n}^*] = \sigma_1^2 \delta_n. \end{aligned}$$

The above autocorrelation is not periodic, as, for example,

$$a_{y,2k,n+2} = E[y_{2k} y_{2k-n-2}^*] = \sigma_0^2 \delta_{n+2} \neq a_{y,2k,n}.$$

The sequence  $y$  is, however, WSCS with period 2 (easy to check from (2.243)).

<sup>63</sup>While we will examine the effects of downsamplers and upsamplers in what follows, the example is simple enough that we can go through it now.



**Figure 2.28:** Generative model for for a  $WSCS_2$  sequence. The input  $x$  is WSS, while the output  $y$  is  $WSCS_2$  (Example 2.35).

As we have done earlier for deterministic sequences, we can characterize vector processes using autocorrelation matrices. For the sake of simplicity, let us consider a WSS sequence  $x$  and the vector of its polyphase components as in (2.213). Then, its matrix autocorrelation will be

$$\begin{aligned}
 A_n &= \begin{bmatrix} a_{0,n} & c_{01,n} \\ c_{10,n} & a_{1,n} \end{bmatrix} \\
 &= \begin{bmatrix} E \begin{bmatrix} x_{0,k} & x_{0,k-n}^* \end{bmatrix} & E \begin{bmatrix} x_{0,k} & x_{1,k-n}^* \end{bmatrix} \\ E \begin{bmatrix} x_{1,k} & x_{0,k-n}^* \end{bmatrix} & E \begin{bmatrix} x_{1,k} & x_{1,k-n}^* \end{bmatrix} \end{bmatrix} \\
 &\stackrel{(a)}{=} \begin{bmatrix} E \begin{bmatrix} x_{2k} & x_{2k-2n}^* \end{bmatrix} & E \begin{bmatrix} x_{2k} & x_{2k-2n+1}^* \end{bmatrix} \\ E \begin{bmatrix} x_{2k+1} & x_{2k-2n}^* \end{bmatrix} & E \begin{bmatrix} x_{2k+1} & x_{2k-2n+1}^* \end{bmatrix} \end{bmatrix} \\
 &\stackrel{(b)}{=} \begin{bmatrix} a_{2n} & a_{2n-1} \\ a_{2n+1} & a_{2n} \end{bmatrix}, \tag{2.244}
 \end{aligned}$$

where (a) follows from the definition of the polyphase components of  $x$  and (b) from  $x$  being WSS. In the DTFT domain:

$$A(e^{j\omega}) = \begin{bmatrix} A_0(e^{j\omega}) & e^{-j\omega} A_1(e^{j\omega}) \\ A_1(e^{j\omega}) & A_0(e^{j\omega}) \end{bmatrix}, \tag{2.245}$$

where  $A(e^{j\omega})$  is the power spectral density of  $x$ , and  $A_0(e^{j\omega})$  and  $A_1(e^{j\omega})$  are the DTFTs of the polyphase components of  $a_n$ . The matrix  $A(e^{j\omega})$  is positive semidefinite, which we now prove. For simplicity, we assume real entries. First, we know that  $A(e^{j\omega})$  is an even function of  $\omega$ , (2.97c), and nonnegative, (2.97b). Furthermore,

$$\begin{aligned}
 A_0(e^{2j\omega}) &= \frac{1}{2}[A(e^{j\omega}) + A(e^{j(\omega+\pi)})] = A_0(e^{-2j\omega}), \\
 A_1(e^{2j\omega}) &= e^{j\omega} \frac{1}{2}[A(e^{j\omega}) - A(e^{j(\omega+\pi)})] = e^{-2j\omega} A_1(e^{-2j\omega}). \tag{2.246}
 \end{aligned}$$

We thus get that

$$\begin{aligned}
 A_0(e^{2j\omega}) + e^{-j\omega} A_1(e^{2j\omega}) &= A(e^{j\omega}) \stackrel{(a)}{\geq} 0, \\
 A_0(e^{2j\omega}) - e^{-j\omega} A_1(e^{2j\omega}) &= A(e^{j(\omega+\pi)}) \stackrel{(b)}{\geq} 0, \tag{2.247}
 \end{aligned}$$

where (a) follows from (2.97b) and (b) from (2.97a)–(2.97b). Then,

$$\begin{aligned} & (A_0(e^{2j\omega}) + e^{-j\omega} A_1(e^{2j\omega}))(A_0(e^{2j\omega}) - e^{-j\omega} A_1(e^{2j\omega})) \\ &= A_0^2(e^{2j\omega}) - e^{-2j\omega} A_1^2(e^{2j\omega}) \geq 0. \end{aligned} \quad (2.248)$$

To prove that  $A(e^{j\omega})$  is positive semidefinite, we must prove that all of its principal minors have nonnegative determinants (see Section 1.B.2). In this case, that means  $A_0(e^{j\omega}) \geq 0$ , which we know from (2.97b), as well as

$$\det(A(e^{j\omega})) = A_0^2(e^{j\omega}) - e^{-j\omega} A_1^2(e^{j\omega}) \stackrel{(a)}{\geq} 0,$$

where (a) follows from (2.248), proving that  $A(e^{j\omega})$  is positive semidefinite.

**EXAMPLE 2.36 (FIRST-ORDER AR PROCESS, EXAMPLE 2.32 CONT'D)** We illustrate this with an example. The proof follows a different path from the one we have just seen and is more explicit. As a reminder, we are looking at a WSS sequence  $x$  and the vector of its polyphase components as in (2.213). We take the input to be an AR-1 process  $x$  with power spectral density as in (2.234), which we know is positive definite when  $|a| < 1$ . The matrix  $A(e^{j\omega})$  is then

$$A(e^{j\omega}) = \frac{1 - a^4}{(1 - a^2 e^{-j\omega})(1 - a^2 e^{j\omega})} \begin{bmatrix} 1 & \frac{a}{1+a^2}(1 + e^{j\omega}) \\ \frac{a}{1+a^2}(1 + e^{-j\omega}) & 1 \end{bmatrix}. \quad (2.249)$$

To check that the matrix is positive definite, we compute  $y^T A(e^{j\omega}) y$ , for an arbitrary  $y = [\cos \theta \quad \sin \theta]$ :

$$[\cos \theta \quad \sin \theta] A(e^{j\omega}) \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} = (1 - a^2) \frac{(1 + a^2) + a(1 + \cos \omega) \sin 2\theta}{1 + a^4 - 2a^2 \cos \omega}.$$

The denominator of the above expression is always positive since

$$1 + a^4 - 2a^2 \cos \omega \stackrel{(a)}{\geq} 1 + a^4 - 2a^2 = (1 - a^2)^2 > 0,$$

where (a) follows from  $|a| < 1$ . Then, we just have to show that the following expression is nonnegative:

$$\begin{aligned} (1 + a^2) + a(1 + \cos \omega) \sin 2\theta & \stackrel{(a)}{\geq} 1 + a^2 + a \sin 2\theta \\ & \stackrel{(b)}{\geq} 1 + a^2 - 2|a| \\ & \stackrel{(c)}{=} (1 - |a|)^2 > 0, \end{aligned}$$

where (a) follows from  $(1 + \cos \omega) \geq 0$ ; (b) from  $a \sin 2\theta \geq |a|$ ; and (c) from  $|a| < 1$ .

We could have saved ourselves all this computation had we observed that

$$A(e^{j\omega}) = U(e^{j\omega}) U^T(e^{-j\omega}), \quad (2.250a)$$

## 2.8. Discrete Stochastic Processes and Systems

279

with

$$U(e^{j\omega}) = \frac{\sqrt{1-a^2}}{1-a^2e^{-j\omega}} \begin{bmatrix} 1 & a \\ ae^{-j\omega} & 1 \end{bmatrix}, \quad (2.250b)$$

making it obvious that  $A(e^{j\omega})$  is positive semidefinite.

**EXAMPLE 2.37 (FIRST-ORDER MA PROCESS, EXAMPLE 2.33 CONT'D)** The matrix autocorrelation is

$$\begin{aligned} A(e^{j\omega}) &= \begin{bmatrix} A_0(e^{j\omega}) & A_1(e^{j\omega}) \\ A_1^*(e^{-j\omega}) & A_0(e^{j\omega}) \end{bmatrix} = \begin{bmatrix} 1 & (1+e^{j\omega})/2 \\ (1+e^{-j\omega})/2 & 1 \end{bmatrix} \\ &= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & e^{j\omega} \\ 1 & 1 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ e^{-j\omega} & 1 \end{bmatrix} = U(e^{j\omega}) U^T(e^{-j\omega}), \end{aligned}$$

and is thus clearly positive semidefinite since it affords a spectral factorization.

We now go through some basic results involving multirate components with WSS or WSCS inputs; a summary is given in Table 2.10.

System	Input $x$	Output $y$	
Downsampling by $N$	WSS	WSS	
	WSCS $_N$	WSS	
	WSCS $_M$	WSCS $_L$	$L = M/\gcd(M, N)$
Upsampling by $N$	WSS	WSCS $_N$	
Filtering by $h$	WSS	WSS	
	WSCS $_N$	WSCS $_N$	
Filtering followed by downsampling	WSS	WSS	
Upsampling followed by filtering	WSS	WSCS $_N$	
	WSS	WSS	filter has alias-free support
Rational change ( $M$ up $N$ down)	WSS	WSCS $_L$	$L = M/\gcd(M, N)$

**Table 2.10:** Summary of results for multirate systems with stochastic inputs.

**Downsampling** Given is a downsampler by  $N$  as in (2.183). Then, if  $x$  is WSS,  $y$  is WSS as well:

$$a_{y,k,n} = E[y_k y_{k-n}^*] \stackrel{(a)}{=} E[x_{Nk} x_{N(k-n)}^*] \stackrel{(b)}{=} a_{x,Nn} = a_{y,n}, \quad (2.251)$$

where (a) follows from (2.183), and (b) from  $x$  being WSS.

If  $x$  is WSCS $_N$ , then  $y$  is WSS as well,

$$a_{y,k,n} = E[y_k y_{k-n}^*] \stackrel{(a)}{=} E[x_{Nk} x_{N(k-n)}^*] \stackrel{(b)}{=} a_{x,Nn} = a_{y,n},$$

where again, (a) follows from (2.183); and (b) from  $x$  being WSCS $_N$ .

The above two cases are special cases of the more general fact that if  $x$  is  $\text{WSCS}_M$ , then  $y$  is  $\text{WSCS}_L$ , with  $L = M/\gcd(M, N)$ . For this, and other special cases, see *Further Reading*.

In the DTFT domain, the power spectral density of the output is given by

$$A_y(e^{j\omega}) \stackrel{(a)}{=} \sum_{n \in \mathbb{Z}} a_{y,n} e^{-j\omega n} \stackrel{(b)}{=} \sum_{n \in \mathbb{Z}} a_{x, Nn} e^{-j\omega n} \stackrel{(c)}{=} \frac{1}{N} \sum_{k=0}^{N-1} A_x(e^{j(\omega - 2\pi k)/N}), \quad (2.252)$$

where (a) follows from the definition of the power spectral density (2.232); (b) from (2.251); and (c) from the expression for downsampling by  $N$ , (2.184).

**Upsampling** Given is an upsampler by  $N$  as in (2.188). Then, if  $x$  is WSS,  $y$  is  $\text{WSCS}_N$ . This is easily seen if we remember Definition 2.18:  $y$  will be  $\text{WSCS}_N$  if all of its polyphase components are WSS. All polyphase components of  $y$ , except for the first one, are zero, and are thus WSS. The first polyphase component is just the input sequence  $x$ , which is WSS by assumption.

In the DTFT domain, the power spectral density of the output is given by

$$A_y(e^{j\omega}) = A_x(e^{jN\omega}), \quad (2.253)$$

from the expression for upsampling by  $N$ , (2.189).

**Filtering** We need one more element, a filter, to be able to build basic multirate systems. The following holds: given a  $\text{WSCS}_N$  input sequence  $x$ , and a LPSV system with period  $N$ , the output  $y$  will also be  $\text{WSCS}_N$ . The proof is straightforward:

$$\begin{aligned} a_{y,n+N} &= \mathbb{E}[y_k y_{k-(n+N)}^*] \stackrel{(a)}{=} \mathbb{E}\left[\sum_m h_m x_{k-m} \sum_\ell h_\ell^* x_{k-(n+N)-\ell}^*\right] \\ &\stackrel{(b)}{=} \sum_{m,\ell} h_m h_\ell^* \mathbb{E}[x_{k-m} x_{k-(n+N)-\ell}^*] \stackrel{(c)}{=} \sum_{m,\ell} h_m h_\ell^* a_{x,n+N-m+\ell} \\ &\stackrel{(d)}{=} \sum_{m,\ell} h_m h_\ell^* a_{x,n-m+\ell} = \sum_{m,\ell} h_m h_\ell^* \mathbb{E}[x_{k-m} x_{k-n-\ell}^*] \\ &= \mathbb{E}\left[\sum_m h_m x_{k-m} \sum_\ell h_\ell^* x_{k-n-\ell}^*\right] = \mathbb{E}[y_k y_{k-n}^*] = a_{y,n}, \end{aligned}$$

where (a) follows from the convolution expression (2.58); (b) from linearity of expectation and  $h$  being deterministic; (c) from the expression for the autocorrelation of  $x$ , (2.223); and (d) from  $x$  being  $\text{WSCS}_N$ , (2.243b). For the DTFT-domain expression of the autocorrelation of a filtered sequence, see (2.236).

**Filtering Followed by Downsampling** Finally, we put together some simple combinations of the basic operations we have seen so far. Start with filtering followed by downsampling as in Figure 2.19. We have already seen that downsampling does



not change the nature of the sequence, nor does filtering. Thus, if  $x$  is WSS,  $y$  is WSS as well. In the DTFT domain, using (2.236) and (2.252), we get

$$A_y(e^{j\omega}) = \frac{1}{N} \sum_{k=0}^{N-1} A_{\tilde{g}}(e^{j(\omega-2\pi k)/N}) A_x(e^{j(\omega-2\pi k)/N}), \quad (2.254)$$

where  $A_{\tilde{g}}(e^{j\omega}) = |\tilde{G}(e^{j\omega})|^2$  is the DTFT of the deterministic autocorrelation of  $\tilde{g}$ .

**Upsampling Followed by Filtering** We now look at upsampling followed by filtering, as shown in Figure 2.20. Then, if  $x$  is WSS,  $y$  is  $\text{WSCS}_N$ . To see that, we use what we have shown so far. We know that if  $x$  is WSS, the output of the upsampler will be  $\text{WSCS}_N$ . As this is the input to an LSI (and consequently LPSV) system, we have shown that its output will be  $\text{WSCS}_N$ . We illustrate this with an example:

**EXAMPLE 2.38 (UPSAMPLING AND FILTERING, EXAMPLE 2.27 CONT'D)** Given is a AWGN sequence  $x$  as the input to the system in Figure 2.20, with  $g_n = \delta_n + \delta_{n-1}$  as in Example 2.27. After upsampling, the first polyphase component is just  $x$  itself, so is WSS, while the second polyphase component is all zero, and is thus WSS as well. The output is not WSS since clearly (2.224) is not satisfied. The output of the upsampler is thus  $\text{WSCS}_2$ . After filtering, the output is the staircase sequence from (2.200). This discrete stochastic process is not WSS, but is  $\text{WSCS}_2$  as each of its two polyphase components are now equal to  $x$ .

In fact, an important result is that a necessary and sufficient condition for the output of upsampling and filtering to be WSS for a WSS input is for the filter  $g$  to have alias-free support. One could think of this as an ideal  $N$ th-band filter, or, more generally, a filter that would extract pieces of the repeated contracted input spectrum from Figure 2.18(c) so as to reconstruct one full input spectrum.

In the DTFT domain, using (2.236) and (2.253), we get

$$A_y(e^{j\omega}) = A_g(e^{j\omega}) A_x(e^{jN\omega}). \quad (2.255)$$

**Rational Sampling Rate Change** Suppose now we have a combination of upsampling by  $M$ , followed by filtering, followed by downsampling by  $N$ . We can say that if  $x$  is WSS, then  $y$  is  $\text{WSCS}_L$ , with  $L = M/\text{gcd}(M, N)$ . This follows directly from the fact we just proved on upsampling followed by filtering and applying the result on downsampling.

## 2.9 Computational Aspects

In this section we give an overview of FFT algorithms and their direct application to computation of circular convolutions, as well as linear convolutions. We then discuss the complexity of some multirate operations.

### 2.9.1 Fast Fourier Transforms

We have thus far studied the DFT as the natural analysis tool for either periodic sequences, or finite-length sequences circularly extended. We now study fast algorithms for computing the DFT called *fast Fourier transform (FFT)* algorithms.

The DFT, defined in (2.159a), is a sum of  $N$  complex terms, each the product of a complex constant (a power of  $W_N$ ) and a component of the input sequence  $x$  (vector-vector product). Thus, each DFT coefficient can be computed with  $N$  multiplications and  $(N-1)$  additions.<sup>64</sup> There are  $N$  such vector-vector products, and thus, the full DFT can be computed with  $\mu = N^2$  multiplications and  $\nu = N(N-1)$  additions, for a total cost of

$$C_{\text{DFT,direct}} = N(N-1) + N^2 = 2N^2 - N \sim O(N^2), \quad (2.256)$$

exactly the cost of a direct matrix-vector multiplication of an  $N \times N$  DFT matrix  $F_N$  by an  $N \times 1$  vector  $X$ , as in (1.166b).

**Radix-2 FFT** The special structure of the DFT allows its computation with far fewer operations than  $O(N^2)$ , based on decomposing it into smaller DFTs and a few simple operations to combine the results. For illustration purposes, we consider in detail only  $N = 2^k$ ,  $k \in \mathbb{Z}^+$ , and only briefly comment on fast algorithms for other values of  $N$ .

Starting with the definition of the DFT (2.159a), write

$$\begin{aligned} X_k &= \sum_{n=0}^{N-1} x_n W_N^{kn} \stackrel{(a)}{=} \sum_{n=0}^{N-1} x_{2n} W_N^{k(2n)} + \sum_{n=0}^{N-1} x_{2n+1} W_N^{k(2n+1)} \\ &\stackrel{(b)}{=} \sum_{n=0}^{N-1} x_{2n} W_{N/2}^{kn} + W_N^k \sum_{n=0}^{N-1} x_{2n+1} W_{N/2}^{kn}, \end{aligned} \quad (2.257)$$

where (a) separates the summation over odd- and even-numbered terms; and (b) follows from  $W_N^2 = W_{N/2}$ . Recognize the first sum as the length- $N/2$  DFT of the sequence  $[x_0 \ x_2 \ \dots \ x_{N-2}]^T$ , and the second sum as the length- $N/2$  DFT of the sequence  $[x_1 \ x_3 \ \dots \ x_{N-1}]^T$ . It is now apparent that the length- $N$  DFT computation can make use of  $F_{N/2} D_2 x$  and  $F_{N/2} C_2 x$ , where  $D_2$  is the downsampling-by-2 operator defined in (2.180b), and  $C_2$  is a similar operator, except that it keeps the odd-indexed values. Since the length- $N/2$  DFT is  $(N/2)$ -periodic in  $k$ , (2.257), can be used both for  $k \in \{0, 1, \dots, N/2-1\}$  as well as  $k \in \{N/2, N/2+1, \dots, N-1\}$ .

To get a compact matrix representation, we introduce the diagonal matrix:

$$A_{N/2} = \text{diag}([1, W_N, W_N^2, \dots, W_N^{(N/2)-1}]),$$

<sup>64</sup>We are counting *complex* multiplications and *complex* additions. It is customary to not count multiplications by  $(-1)$  and thus lump together additions and subtractions.

## 2.9. Computational Aspects

283

and rewrite (2.257) as

$$\begin{bmatrix} X_0 \\ \vdots \\ X_{N/2-1} \end{bmatrix} = F_{N/2} D_2 x + A_{N/2} F_{N/2} C_2 x, \quad (2.258a)$$

$$\begin{bmatrix} X_{N/2} \\ \vdots \\ X_{N-1} \end{bmatrix} = F_{N/2} D_2 x - A_{N/2} F_{N/2} C_2 x, \quad (2.258b)$$

where the final twist was to realize that  $W_N^k = -W_N^{k-N/2}$ , leading to

$$X = F_N x = \begin{bmatrix} I_{N/2} & A_{N/2} \\ I_{N/2} & -A_{N/2} \end{bmatrix} \begin{bmatrix} F_{N/2} & 0 \\ 0 & F_{N/2} \end{bmatrix} \begin{bmatrix} D_2 \\ C_2 \end{bmatrix} x. \quad (2.259)$$

If this turns out to be a useful factorization, then we can repeat it to represent  $F_{N/2}$  using  $F_{N/4}$ , etc, until we reach  $F_2$ , which requires no multiplications,  $\mu_2 = 0$ , and only  $\nu_2 = 2$  additions. Let us count computations in the factored form.

With  $\mu_N$  and  $\nu_N$  the number of multiplications and additions in computing a length- $N$  DFT, the factorization (2.259) shows that a length- $N$  DFT can be computed using two length- $N/2$  DFTs,  $N/2$  multiplications and  $N$  additions. Iterating on the length- $N/2$  DFTs, then length- $N/2$  DFTs, and so on, leads to the following recursions:

$$\begin{aligned} \nu_N &= 2\nu_{N/2} + N = 2^2\nu_{N/2^2} + 2N = 2^3\nu_{N/2^3} + 2^2N = \dots \\ &= \frac{N}{2}\nu_2 + (\log_2 N - 1)N = N \log_2 N, \\ \mu_N &= 2\mu_{N/2} + \frac{N}{2} = 2^2\mu_{N/2^2} + 2\frac{N}{2} = 2^2\mu_{N/2^3} + 2^2\frac{N}{2} = \dots \\ &= \frac{N}{2}\mu_2 + (\log_2 N - 1)\frac{N}{2} = \frac{N}{2} \log_2 N - \frac{N}{2}, \\ C_{\text{DFT,radix-2}} &= \frac{3}{2}N \log_2 N - \frac{N}{2} \sim O(N \log_2 N). \end{aligned} \quad (2.260)$$

Thus, recursive application of (2.259) reduces the cost from  $O(N^2)$  in (2.256) to  $O(N \log_2 N)$  in (2.260). We illustrate the above procedure with a simple example:

**EXAMPLE 2.39 (COMPUTATION OF THE LENGTH-4 FFT)** We check that the fac-

torization (2.259) indeed equals the length-4 DFT:

$$\begin{aligned}
 F_4 &= \begin{bmatrix} I_2 & A_2 \\ I_2 & -A_2 \end{bmatrix} \begin{bmatrix} F_2 & 0 \\ 0 & F_2 \end{bmatrix} \begin{bmatrix} D_2 \\ C_2 \end{bmatrix} \\
 &= \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & j \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -j \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W_4 & W_4^2 & W_4^3 \\ 1 & W_4^2 & W_4^4 & W_4^6 \\ 1 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix},
 \end{aligned}$$

exactly (2.161a). We can also write out (2.257):

$$\begin{aligned}
 X_k &= \sum_{n=0}^3 x_n W_4^{kn} = \sum_{n=0}^3 x_{2n} W_4^{k(2n)} + \sum_{n=0}^3 x_{2n+1} W_4^{k(2n+1)} \\
 &= \sum_{n=0}^3 x_{2n} W_2^{kn} + W_4^k \sum_{n=0}^3 x_{2n+1} W_2^{kn} \\
 &= (x_0 W_2^0 + x_2 W_2^k) + W_4^k (x_1 W_2^0 + x_3 W_2^k) \\
 &= (x_0 + (-1)^k x_2) + W_4^k (x_1 + (-1)^k x_3),
 \end{aligned}$$

equivalent to computing one DFT of length 2 on even samples, then one DFT of length 2 on odd samples, and finally multiplying those by a constant  $W_4^k$ .

**Other FFT Algorithms** A more general, and the most famous, class of FFT algorithms, the *Cooley-Tukey FFT*, works for any composite length  $N = N_1 N_2$ . The algorithm breaks down a length- $N$  DFT into  $N_2$  length- $N_1$  DFT,  $N$  complex factors and  $N_1$  length- $N_2$  DFT. Often, either  $N_1$  or  $N_2$  is a small factor called a *radix*.

The *Good-Thomas FFT* works for  $N = N_1 N_2$  where  $N_1$  and  $N_2$  are coprime. It is based on the Chinese remainder theorem and avoids the complex factors of the Cooley-Tukey FFT. Thus, the algorithm breaks down a length- $N$  DFT into  $N_2$  length- $N_1$  DFT and  $N_1$  length- $N_2$  DFT, equivalent to a two-dimensional length- $(N_1 \times N_2)$  DFT.

The *Rader's FFT* works for prime length  $N$ . It is based on mapping the computation of the DFT into a computation of a circular convolution of length  $N-1$  (recall that (2.177) shows that the DFT diagonalizes the circulant convolution operator). Winograd extended the Rader's FFT to include powers of prime lengths, and is sometimes considered as a subclass of the Winograd FFT which we discuss next.

The *Winograd FFT* is often used for small factors. It is based on considering a  $N_1$  length- $N_1 N_2$  DFT as a two-dimensional DFT of length  $(N_1 \times N_2)$ , as we have seen for the Good-Thomas algorithm. If  $N_1$  and  $N_2$  are prime, we can use the Rader's FFT on each  $N_1$  and  $N_2$ . While it is less costly in terms of required additions and multiplications, it is also complex and thus not often used.

The *split-radix FFT* is used for  $N$  that are multiples of 4 and recursively splits length- $N$  DFT in terms of one length- $N/2$  DFT and two length- $N/4$  DFTs and boasts the lowest operations count for  $N = 2^k$ ,  $k > 1$ .

Remember that the cost in terms of additions and multiplications is just one measure of how fast an algorithm can be computed; many other factors come into play including the specific computing platform. As a result, the Cooley-Tukey FFT is still the prevalent one, despite some of the other algorithms having a lower multiplication, addition, or a total operations count. Moreover, as we have just seen, there exists an FFT for any given  $N$ , and each of them is  $O(N \log_2 N)$ , carrying the cost of

$$C_{\text{DFT,FFT}} = \alpha N \log_2 N \quad \sim \quad O(N \log_2 N), \quad (2.261)$$

where  $\alpha$  is a small constant dependent on the actual FFT algorithm.

### 2.9.2 Convolution

In discussing convolution, for the first time we will encounter the issue of computing on a finite-length input (offline computation), or, an infinite-length one (online computation). In the first case, we will be computing the cost per block of input samples, while in the second, we will be computing the cost per input sample.

**Computing Circular Convolution** Since the DFT operator diagonalizes circular convolution, it is easy to estimate its cost. Using (2.177), computing  $H$  requires a length- $N$  DFT,  $N$  pointwise multiplications by the diagonal matrix entries and a length- $N$  inverse DFT. Using (2.261) for the cost of the DFT, computing the circular convolution carries the cost of

$$C_{\text{conv,freq}} = 2\alpha N \log_2 N + N \quad \sim \quad O(N \log_2 N), \quad (2.262)$$

per  $N$  input samples.

**Computing Linear Convolution** We start with a straight implementation of linear convolution in time domain (2.58) for a finite-length input  $x$ . Without loss of generality, assume that the input is of length  $M$  and the filter  $h$  is of length  $L < M$ . We need 1 multiplication and 0 additions to compute  $y_0 = h_0 x_0$ , 2 multiplications and 1 additions for  $y_1 = h_1 x_0 + h_0 x_1$ , all the way to  $L$  multiplications and  $(L-1)$  additions for  $y_{L-1} = \sum_{k=0}^{L-1} x_k h_{L-1-k}$ ,  $y_L, \dots, y_{M-1}$ , and then back down to 1 multiplication and 0 additions for  $y_{M+L-1} = h_{L-1} x_{M-1}$ , leading to

$$\begin{aligned} C_{\text{conv,time}} &= \nu + \mu \\ &= \left[ 2 \sum_{k=1}^{L-2} k + (M-L+1)(L-1) \right] + \left[ 2 \sum_{k=1}^{L-1} k + (M-L+1)L \right] \\ &= 2 \frac{(L-2)(L-1)}{2} + 2 \frac{(L-1)L}{2} + (M-L+1)(2L-1) \\ &= 2ML - M - L + 1 \quad \sim \quad O(ML), \end{aligned} \quad (2.263)$$

per  $M$  input samples.

In (2.262), we saw a very efficient implementation of the circular convolution using FFTs. We will now show how to reduce the problem of computing the linear convolution of an infinite-length input to that of computing the circular convolution, and use the results we just developed. To that end, we build upon Example 2.13 on the equivalence of circular and linear convolutions.

EXAMPLE 2.40 (EQUIVALENCE OF CIRCULAR AND LINEAR CONVOLUTIONS, EXAMPLE 2.13 CONT'D)

In Example 2.13, we wanted to compute the linear convolution of a length-3 filter with a length-4 sequence, and found that computing a circular convolution instead was equivalent. We rewrite (2.74b) as follows:

$$\begin{aligned}
 \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} &= \begin{bmatrix} h_0 & 0 & 0 & 0 & h_2 & h_1 \\ h_1 & h_0 & 0 & 0 & 0 & h_2 \\ h_2 & h_1 & h_0 & 0 & 0 & 0 \\ 0 & h_2 & h_1 & h_0 & 0 & 0 \\ 0 & 0 & h_2 & h_1 & h_0 & 0 \\ 0 & 0 & 0 & h_2 & h_1 & h_0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ 0 \\ 0 \end{bmatrix} \\
 &= \begin{bmatrix} h_0 & 0 & 0 & 0 & h_2 & h_1 \\ h_1 & h_0 & 0 & 0 & 0 & h_2 \\ h_2 & h_1 & h_0 & 0 & 0 & 0 \\ 0 & h_2 & h_1 & h_0 & 0 & 0 \\ 0 & 0 & h_2 & h_1 & h_0 & 0 \\ 0 & 0 & 0 & h_2 & h_1 & h_0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = H_6 \begin{bmatrix} I_4 \\ 0_{2 \times 4} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix},
 \end{aligned}$$

where the input vector is now stated explicitly without the trailing zeros,  $H_6$  is the  $6 \times 6$  circulant matrix as in (2.73),  $I_4$  is a  $4 \times 4$  identity matrix, and  $0_{2 \times 4}$  is a  $2 \times 4$  all zero matrix.

Imagine now that instead of a finite-length input sequence  $x$ , we have an infinite one. A way to compute a convolution of  $h$  with  $x$  is to break  $x$  into pieces of length 4, and then repeat the above procedure. The only issue is that because  $x$  is of infinite length, only outputs  $y_2$  and  $y_3$  are correct; for example,  $y_0$  is not correct as it requires  $x_{-1}$  and  $x_{-2}$  to compute it, and in the above, we assumed that  $x_{-1} = x_{-2} = 0$ . Let us look at two blocks of outputs for two consecutive blocks of inputs,  $[x_{-4} \ x_{-3} \ x_{-2} \ x_{-1}]^T$  and  $[x_0 \ x_1 \ x_2 \ x_3]^T$ , and denote those outputs that are only partially computed by  $^{(p-)}$  (for the first block) and

## 2.9. Computational Aspects

287

$^{(p+)}$  (for the second block),

$$\begin{array}{rclcl}
 y_{-4}^{(p-)} & = & & & h_0 x_{-4}] \\
 y_{-3}^{(p-)} & = & & h_1 x_{-4} & + & h_0 x_{-3}] \\
 y_{-2} & = & h_2 x_{-4} & + & h_1 x_{-3} & + & h_0 x_{-2}] \\
 y_{-1} & = & h_2 x_{-3} & + & h_1 x_{-2} & + & h_0 x_{-1}] \\
 y_0^{(p-)} & = & h_2 x_{-2} & + & h_1 x_{-1}] & [h_0 x_0 & = & y_0^{(p+)} \\
 y_1^{(p-)} & = & h_2 x_{-1}] & [h_1 x_0 & + & h_0 x_1 & = & y_1^{(p+)} \\
 & & [h_2 x_0 & + & h_1 x_1 & + & h_0 x_2 & = & y_2 \\
 & & [h_2 x_1 & + & h_1 x_2 & + & h_0 x_3 & = & y_3 \\
 & & [h_2 x_2 & + & h_1 x_3 & & & = & y_4^{(p+)} \\
 & & [h_2 x_3 & & & & & = & y_5^{(p+)}
 \end{array}$$

where on the left we have the first output block and on the right the second output block, and the brackets show to which block the summands belong. What we can see is that the partial outputs in two consecutive blocks complement each other, giving the correct result. For example, adding  $y_0^{(p-)}$  to  $y_0^{(p+)}$  results in  $h_2 x_{-2} + h_1 x_{-1} + h_0 x_0$ , the correct result for  $y_0$ . In other words, we have to offset the second vector down by 4 rows and add them up. We can express this in matrix form as:

$$\begin{bmatrix} \vdots \\ y_{-4} \\ \vdots \\ y_0 \\ \vdots \\ y_5 \\ \vdots \end{bmatrix} = A H E \begin{bmatrix} \vdots \\ x_{-4} \\ \vdots \\ x_{-1} \\ x_0 \\ \vdots \\ x_3 \\ \vdots \end{bmatrix}, \quad \text{with } H = \begin{bmatrix} \ddots & & & & & & \\ & \ddots & & & & & \\ & & H_6 & & & & \\ & & & H_6 & & & \\ & & & & \ddots & & \\ & & & & & \ddots & \\ & & & & & & \ddots \end{bmatrix},$$

and

$$A = \begin{bmatrix} \ddots & & & & & & \\ & I_2 & 0 & 0 & & & \\ & 0 & I_2 & 0 & & & \\ & 0 & 0 & I_2 & I_2 & 0 & 0 \\ & & & & 0 & I_2 & 0 \\ & & & & 0 & 0 & I_2 \\ & & & & & & \ddots \end{bmatrix} \quad E = \begin{bmatrix} \ddots & & & & & & \\ & I_2 & 0 & & & & \\ & 0 & I_2 & & & & \\ & 0 & 0 & & & & \\ & & & I_2 & 0 & & \\ & & & 0 & I_2 & & \\ & & & 0 & 0 & & \\ & & & & & \ddots & \end{bmatrix}.$$

We now see why the computation of the convolution as above would be efficient, since multiplication by  $E$  is merely the insertion of zeros (*extension*), multiplication by  $H$  is circular convolution as we have just seen, and multiplication by  $A$  (*addition*) requires only 2 additions for each block of 4 input samples.

We now generalize what we have seen in this simple example. Given a length- $L$  FIR filter and input vector  $x$  in blocks of  $M$  samples, the above procedure, called the *overlap-save algorithm*, can be viewed as the following factorization:

$$Y = A H E X, \quad (2.264)$$

where  $Y$  is the output vector,  $H$  is a block-diagonal matrix with circular convolution operator  $H_N$  on the diagonal (with  $N = L + M - 1$ , according to Proposition 2.10),  $A$  is the addition matrix with  $I_N$  on the diagonal offset by  $M$  rows, and  $E$  is the extension matrix with  $I_M$  on the diagonal offset by  $N$  rows.

We now compute the cost of the algorithm. As we said,  $E$  merely inserts zeros and thus has no cost;  $H$  costs  $(2\alpha N \log_2 N + N)$  operations according to (2.262); and,  $A$  requires  $(N - M)$  additions, for a total cost of

$$C_{\text{conv,overlap-add}} = \frac{2\alpha}{M} N \log_2 N + \frac{2}{M} N - 1 \quad \sim \quad O(\log_2 N), \quad (2.265)$$

per input sample. In the above, we have not made the substitution  $N = L + M - 1$ , as in practice,  $N$  larger than that necessary minimum is often chosen.

A dual algorithm to the one we just presented is the *overlap-save algorithm*, whose cost is similar with a small advantage of no additions in the final stage (as the factorization is practically the transpose of (2.264)). However, its disadvantage is that the DFTs are calculated on denser input vectors, and thus might make the FFTs more costly.

### 2.9.3 Multirate Operations

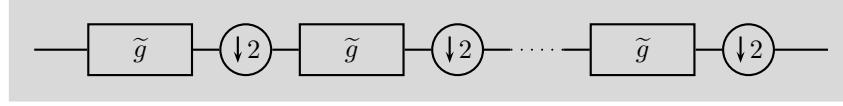
The key to improving the computational efficiency in multirate signal processing is simple: always operate at the lowest possible sampling frequency (rate). We now show this idea in action, bearing in mind that downsampling and upsampling have no cost in terms of additions and multiplications (although they might require memory access).

**Filtering Followed by Downsampling** We start with time-domain computation. Assume we directly compute the convolution  $(h * x)$  of the length- $M$  input  $x$  and length- $L$  filter  $h$ , and simply discard every other sample. Using (2.263), the cost is

$$C_{\text{time,direct}} = 2ML - M - L + 1 \quad \sim \quad O(ML), \quad (2.266)$$

per  $M$  input samples. However, we know better than to waste computations this way. As we have seen before, filtering followed by downsampling by 2 is equivalent to computing only the even samples of the convolution as in (2.220). These polyphase components,  $x_0$  and  $x_1$ , are convolved with the polyphase components of the filter,  $h_1$  and  $h_0$ , which are now of half the length (see Figure 2.25). We thus have to compute two convolutions (2.266) at half the length plus add the results of these convolutions, yielding  $ML - M - L + 2$  operations for the two convolutions





**Figure 2.29:** Cascade of  $K$  filters followed by downsamplers.

plus  $M/2 + L/2 - 1$  additions (since each convolution product is now of length  $M/2 + L/2 - 1$ , for a total cost of

$$C_{\text{time,polyphase}} = ML - \frac{M}{2} - \frac{L}{2} + 1 \sim O(ML), \quad (2.267a)$$

per  $M$  input samples, roughly 50% savings, but still  $O(ML)$ .

We now investigate what happens if we move to frequency domain. According to Proposition 2.10, instead of computing the linear convolution, we can compute a circular convolution with a period of at least  $(M + L - 1)$  in this case, and then discard every other sample as in (2.266). Using a length- $(M + L)$  DFT, according to (2.265)

$$\begin{aligned} C_{\text{freq,direct}} &= 2\alpha(M + L) \log_2(M + L) + M + 2L \\ &\sim O((M + L) \log_2(M + L)), \end{aligned} \quad (2.267b)$$

per  $M$  input samples. If, instead, we use (2.220), we will need two convolutions at half the length plus add the results of these convolutions, for a total cost of

$$\begin{aligned} C_{\text{freq,polyphase}} &= 2\alpha(M + L) \log_2(M + L) + \left(\frac{3}{2} - 2\alpha\right)M + \left(\frac{5}{2} - 2\alpha\right)L - 1 \\ &\sim O((M + L) \log_2(M + L)), \end{aligned}$$

per  $M$  input samples, still  $O((M + L) \log_2(M + L))$  but with savings dependent on the sizes  $M$  and  $L$ . This discussion generalizes straightforwardly to other downsampling factors larger than 2.

**EXAMPLE 2.41 (ITERATION OF FILTERING FOLLOWED BY DOWNSAMPLING)** We have a cascade of  $K$  filters followed by downsamplers as in Figure 2.29. Using any of the expressions we just derived for the cost of one stage  $C$ , we can calculate the cost for  $K$  stages. For the second stage, it is  $C/2$  (because it runs at half the rate of the input), for the third stage it is  $C/4$ , etc, leading to the total cost of

$$C + \frac{C}{2} + \frac{C}{4} + \dots + \frac{C}{2^{K-1}} \stackrel{(a)}{=} \left(2 - \frac{1}{2^{K-1}}\right)C < 2C, \quad (2.268)$$

where (a) follows from (P1.65-1), the formula for a finite geometric series.

**Upsampling Followed by Filtering** The operation of upsampling by 2 followed by filtering in  $z$ -transform domain is expressed via (2.199a). As this operation is dual to the operation of filtering followed by downsampling (they are transposes of each other, see (2.194) and (2.197)), it comes as no surprise that both systems have the same cost.

## Appendix

### 2.A Elements of Analysis

#### 2.A.1 Complex Numbers

The imaginary unit  $j$  is defined as a number that satisfies  $j^2 = -1$ ; since  $j^2 = -1$  implies  $(-j)^2 = -1$ , we are simply fixing one of the two solutions to have the name  $j$ .<sup>65</sup> A complex number  $z \in \mathbb{C}$  is then a number of the form

$$z = a + jb, \quad a, b \in \mathbb{R}. \quad (2.269)$$

In (2.269),  $a$  is called the *real part* while  $b$  is called the *imaginary part*. The *complex conjugate* of  $z$  is denoted by  $z^*$  and is by definition

$$z^* = a - jb. \quad (2.270)$$

Any complex number can be represented in *polar form* as well:

$$z = re^{j\theta}, \quad (2.271)$$

where  $r$  is called the *modulus* or *magnitude* and  $\theta$  is the *argument* or *phase*. Using the *Euler's formula*,

$$e^{j\theta} = \cos \theta + j \sin \theta, \quad (2.272)$$

we can express a complex number further as

$$z = re^{j\theta} = r(\cos \theta + j \sin \theta). \quad (2.273)$$

It allows us to easily find a power of a complex number as

$$(\cos \theta + j \sin \theta)^n = (e^{j\theta})^n = e^{jn\theta} = \cos n\theta + j \sin n\theta. \quad (2.274)$$

Euler's formula highlights that the argument of a complex number is not unique; adding any integer multiple of  $2\pi$  to the argument does not change the number:

$$e^{j\theta+k2\pi} = e^{j\theta}e^{j2k\pi} = e^{j\theta}(e^{j2\pi})^k = e^{j\theta}, \quad \text{for any } k \in \mathbb{Z},$$

since  $e^{j2\pi} = 1$ . Two other useful relations that can be derived using Euler's formula are:

$$\cos \theta = \frac{e^{j\theta} + e^{-j\theta}}{2}, \quad \sin \theta = \frac{e^{j\theta} - e^{-j\theta}}{2j}. \quad (2.275)$$

Complex numbers are typically shown in the complex plane. The complex plane has a one-to-one correspondence with  $\mathbb{R}^2$ , with the real part shown horizontally and the imaginary part vertically. Conversion from polar form  $e^{j\theta}$  to standard (or rectangular) form  $a + jb$  is by

$$a = r \cos \theta, \quad b = r \sin \theta.$$

<sup>65</sup>Mathematicians and physicists typically use  $i$  for the imaginary unit, while  $j$  is more common in engineering.

Conversion from standard form to polar form is simple by just looking at the complex plane, but more complicated to write. One solution is as follows:

$$r = \sqrt{a^2 + b^2}, \quad \theta = \begin{cases} \arctan(b/a), & \text{for } a > 0; \\ \arctan(b/a) + \pi, & \text{for } a < 0, b \geq 0; \\ \arctan(b/a) - \pi, & \text{for } a < 0, b < 0; \\ \pi/2, & \text{for } a = 0, b > 0; \\ -\pi/2, & \text{for } a = 0, b < 0; \\ \text{undefined}, & \text{for } a = 0, b = 0, \end{cases}$$

where  $\arctan$  returns a value in  $(-\pi/2, \pi/2)$ .

Just as a reminder, the basic operations on complex numbers are as follows:

$$\begin{aligned} z_1 + z_2 &= (a_1 + a_2) + j(b_1 + b_2), \\ z_1 - z_2 &= (a_1 - a_2) + j(b_1 - b_2), \\ z_1 z_2 &= (a_1 a_2 - b_1 b_2) + j(b_1 a_2 + a_1 b_2), \\ \frac{z_1}{z_2} &= \frac{a_1 a_2 + b_1 b_2}{a_2^2 + b_2^2} + j \frac{a_2 b_1 - a_1 b_2}{a_2^2 + b_2^2}. \end{aligned}$$

**Roots of Unity** The same way  $j$  was defined as the second root of unity, we can define the principal  $N$ th root of unity as

$$W_N = e^{-j2\pi/N}. \quad (2.276)$$

It is easy to check that  $W_N^k$ , for  $k \in \{2, 3, \dots, N\}$ , are also  $N$ th roots of unity, meaning  $(W_N^k)^N = 1$ . If we drew all  $N$  roots of unity in the complex plane, we would see that they slice up the unit circle by equal angles; the choice of  $W_N$  as the principal root makes  $W_N^0, W_N^1, \dots, W_N^{N-1}$  consecutive in clockwise order. Figure 2.30 shows an example with  $N = 8$ .

Here are some useful identities involving the roots of unity:

$$W_N^N = 1, \quad (2.277a)$$

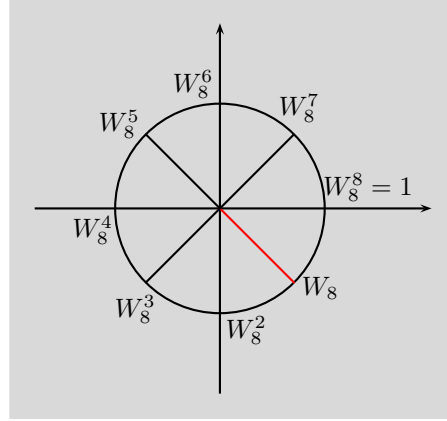
$$W_N^{kN+n} = W_N^n, \quad \text{with } k, n \in \mathbb{Z}, \quad (2.277b)$$

$$\sum_{k=0}^{N-1} W_N^{nk} = \begin{cases} N, & n = \ell N, \ell \in \mathbb{Z}; \\ 0, & \text{otherwise.} \end{cases} \quad (2.277c)$$

The last relation is often referred to as *orthogonality of the roots of unity*. To prove it, for any  $n$  not an integer multiple of  $N$ , use the finite sum formula from (P1.65-1):

$$\sum_{k=0}^{N-1} (W_N^n)^k = \frac{1 - W_N^{Nn}}{1 - W_N^n} = 0, \quad (2.278)$$

since the numerator is 0 and the denominator is nonzero. For  $n = \ell N$ ,  $W_N^{kn} = W_N^{k\ell N} = 1$ , and thus, by direct substitution into (2.277c), we get  $N$ .



**Figure 2.30:** Roots of unity for  $N = 8$  (the principal one is highlighted).

### 2.A.2 Difference Equations

Finding solutions to linear difference equations introduced in Section 2.3.2 involves the following steps:

- (i) *Homogeneous solution:* First, we find a solution to the *homogeneous equation*,

$$y_n^{(h)} = - \sum_{k=1}^N a_k y_{n-k}, \quad (2.279a)$$

by setting the input  $x$  in (2.54) to zero. The solution is of the form

$$y_n^{(h)} = \sum_{k=1}^N \alpha_k \lambda_k^n, \quad (2.279b)$$

where  $\lambda_k$  are obtained by solving the characteristic equation of the system,

$$\sum_{k=0}^N a_k \lambda^{N-k} = 0, \quad (2.279c)$$

and we assumed that  $a_0 = 1$ .

- (ii) *Particular solution:* Then, any particular solution to (2.54),  $y_n^{(p)}$ , is found (independent of  $y_n^{(h)}$ ). This is typically done by assuming that  $y_n^{(p)}$  is of the same form as  $x_n$ , possibly scaled.
- (iii) *Complete solution:* By superposition, the complete solution  $y_n$  is the sum of the homogeneous solution  $y_n^{(h)}$  and the particular solution  $y_n^{(p)}$ :

$$y_n = y_n^{(h)} + y_n^{(p)} = \sum_{k=1}^N \alpha_k \lambda_k^n + y_n^{(p)}. \quad (2.279d)$$

We determine the coefficients  $\alpha_k$  in  $y_n^{(h)}$  by specifying initial conditions for  $y_n$  and then solving the system.

### 2.A.3 Convergence of the Convolution Sum

Recall from Appendix 1.A.2 that a doubly-infinite sum is said to converge when it converges absolutely. Thus, the convolution  $h * x$  between sequences  $h$  and  $x$  is well defined when the sum  $\sum_{k \in \mathbb{Z}} x_k h_{n-k}$  converges absolutely for every value of  $n$ .

When  $h$  and  $x$  are in  $\ell^2(\mathbb{Z})$ , convergence is guaranteed by the fact that the standard  $\ell^2$  inner product is defined on all of  $\ell^2(\mathbb{Z})$  (see Exercise 1.12). Specifically, for each  $n \in \mathbb{Z}$ , define the sequence  $\tilde{h}^{(n)}$  by

$$\tilde{h}_k^{(n)} = h_{n-k}^* \quad \text{for all } k \in \mathbb{Z}.$$

Then each  $\tilde{h}^{(n)}$  is in  $\ell^2(\mathbb{Z})$  because time reversal, shifting, and conjugation do not change the  $\ell^2$  norm. Thus, for every  $n \in \mathbb{Z}$ , the inner product  $\langle x, \tilde{h}^{(n)} \rangle = (h * x)_n$  is well defined, so the convolution is well defined.

In signal processing, we are not quite satisfied with restricting attention to sequences in  $\ell^2(\mathbb{Z})$ ; for example, simple sequences like constants and sinusoids are not in  $\ell^2(\mathbb{Z})$ . To ensure convergence of the convolution sum while loosening the constraints on  $x$  requires tightening the constraints on  $h$ , or vice versa. Hölder's inequality for sequences (1.201) gives a simple condition: The convolution sum is guaranteed to converge absolutely when  $h \in \ell^p(\mathbb{Z})$  and  $x \in \ell^q(\mathbb{Z})$  with  $p$  and  $q$  in  $[1, \infty]$  satisfying  $1/p + 1/q \geq 1$ .<sup>66</sup>

We employ the  $p = 1, q = \infty$  case often: By restricting an LSI system impulse response  $h$  to  $\ell^1(\mathbb{Z})$ , we can allow the input  $x$  to be any sequence in  $\ell^\infty(\mathbb{Z})$  (that is, merely bounded) while ensuring that the output sequence  $h * x$  is well defined. Note that the condition  $h \in \ell^1(\mathbb{Z})$  was already used in BIBO stability of the LSI system with impulse response  $h$  and is a common assumption.

## 2.B Elements of Algebra

### 2.B.1 Polynomials

A *polynomial* is a finite sum of the following form:

$$p(t) = \sum_{n=0}^N a_n t^n. \quad (2.280)$$

Assuming  $a_N \neq 0$ , the *degree of the polynomial* is  $N$ . The set of polynomials with coefficients  $a_n$  from a given ring, themselves form a ring.<sup>67</sup>

<sup>66</sup>Hölder's inequality is given with  $p$  and  $q$  as Hölder conjugates,  $1/p + 1/q = 1$ , and making  $p$  or  $q$  smaller makes the corresponding sequence space smaller; see (1.37).

<sup>67</sup>A ring is a set together with two binary operations, addition and multiplication. The addition operation must be commutative and associative, while the multiplication must be associative, and distributive over addition. There exists an additive identity and each element must have an additive inverse. A standard example of a ring is the set of integers,  $\mathbb{Z}$ .

The *roots* of a polynomial are obtained by equating a *polynomial function*  $p(t)$ , a function obtained by evaluating a polynomial  $p(t)$  over a given domain of  $t$ , to zero. The following theorem, formulated by Gauss, is a useful tool in algebra:

**THEOREM 2.19 (FUNDAMENTAL THEOREM OF ALGEBRA)** Every polynomial with complex coefficients of order  $N$  possesses exactly  $N$  complex roots.

Thus, the degree of the polynomial is also the number of complex roots of that polynomial. For example,  $p(t) = a_2 t^2 + a_1 t + a_0$  is a *quadratic* polynomial and has two roots. An *irreducible* quadratic polynomial is a quadratic polynomial with no real roots. For example,  $p(t) = t^2 + 2$  has no roots in real numbers; rather, its roots are complex,  $\pm j\sqrt{2}$ .

This theorem holds only for polynomials in one variable. For a polynomial with real coefficients, we can factor any polynomial into a product of linear factors,  $(t - b_n)$ , and irreducible quadratic factors with real coefficients,  $(t^2 + c_n t + d_n)$ , while for a polynomial with complex coefficients, the factors are all linear,  $(t - z_n)$ ,

$$p(t) = \sum_{n=0}^N a_n t^n = \begin{aligned} & a_n, b_n, c_n, d_n \in \mathbb{R} \\ & a_N \prod_{n=0}^{N-2k-1} (t - b_n) \prod_{n=0}^{k-1} (t^2 + c_n t + d_n), \\ & a_n, z_n \in \mathbb{C} \\ & a_N \prod_{n=0}^{N-1} (t - z_n). \end{aligned} \quad (2.281)$$

Two polynomials  $p(t)$  and  $q(t)$  are called *coprime*, written as  $(p(t), q(t)) = 1$ , when they have no common factors. The *Bezout* identity states that if  $p(t)$  and  $q(t)$  are coprime, there exist two other polynomials  $a(t)$  and  $b(t)$  such that

$$a(t)p(t) + b(t)q(t) = 1, \quad \text{for all } t. \quad (2.282)$$

Euclid's algorithm is a constructive way of finding  $a(t)$  and  $b(t)$  in (2.282).

**Laurent Polynomials** A *Laurent* polynomial is like a polynomial except that negative powers are allowed in (2.280)

$$p(t) = \sum_{n=-M}^N a_n t^n. \quad (2.283)$$

This can be written as

$$p(t) = t^{-M} q(t) \quad \text{with} \quad q(t) = \sum_{n=0}^{N+M} a_n t^n, \quad (2.284)$$

where  $q(t)$  is now just an ordinary polynomial.

**Ratios of Polynomials** A rational function  $r(t)$  is a ratio of two polynomials

$$r(t) = \frac{p(t)}{q(t)} = \frac{\sum_{n=0}^N a_n t^n}{\sum_{n=0}^M b_n t^n}. \quad (2.285)$$

In general, we assume that  $M \geq N$ , as otherwise, we could use polynomial division to write (2.285) as a sum of a polynomial and a ratio of polynomials with the numerator now of order smaller or equal to  $M$ .

Assume that  $p(t)$  and  $q(t)$  are coprime; otherwise, we can cancel common factors and proceed. When  $M = N$ , by Theorem 2.19, (2.285) has  $N$  zeros and  $M$  poles (zeros of the denominator  $q(t)$ ) in the complex plane. When  $M < N$ , there are  $N - M$  additional zeros at  $t = \infty$ . When  $M > N$ , there are  $M - N$  additional poles at  $t = \infty$ , indicating that a rational function has  $\max(M, N)$  poles and zeros, including ones at 0 and  $\infty$ .

**Discrete Polynomials** A *polynomial sequence* is a sequence whose  $n$ th element is a finite sum of the following form:

$$p_n = \sum_{k=0}^N a_k n^k, \quad n \in \mathbb{Z}. \quad (2.286)$$

For example, a *constant* polynomial sequence is of the form  $p_n = a$ , a *linear* polynomial sequence is of the form  $p_n = a_0 + a_1 n$ , and a *quadratic* polynomial sequence is of the form  $p_n = a_0 + a_1 n + a_2 n^2$ . The  $z$ -transform of such a sequence is:

$$P(z) = \sum_{n \in \mathbb{Z}} p_n z^{-n} = \sum_{n \in \mathbb{Z}} \left( \sum_{k=0}^N a_k n^k \right) z^{-n}.$$

When we study wavelets and filter banks, we will be concerned with the moment annihilating/preserving properties of such systems. The following fact will then be of use: Convolution of the polynomial sequence with a differencing filter  $d_n = (\delta_n - \delta_{n-1})$ , or, multiplication of  $P(z)$  by  $D(z) = (1 - z^{-1})$ , reduces the degree of the polynomial by 1, as in

$$\begin{aligned} D(z)P(z) &= (1 - z^{-1}) \sum_n p_n z^{-n} = (1 - z^{-1}) \sum_n \left( \sum_{k=0}^N a_k n^k \right) z^{-n} \\ &= \sum_n \sum_{k=0}^N a_k n^k z^{-n} - \sum_n \sum_{k=0}^N a_k n^k z^{-(n+1)} \\ &= \sum_n \sum_{k=0}^N a_k n^k z^{-n} - \sum_n \sum_{k=0}^N a_k (n-1)^k z^{-n} \\ &= \sum_n \sum_{k=0}^N a_k (n^k - (n-1)^k) z^{-n} \\ &\stackrel{(a)}{=} \sum_n \left( \sum_{k=0}^{N-1} b_k n^k \right) z^{-n} = \sum_n r_n z^{-n}, \end{aligned}$$

where  $r_n$  is a polynomial of degree  $(N - 1)$ , and (a) follows from  $(n^N - (n - 1)^N)$  being a polynomial of degree  $(N - 1)$ . The above process can be seen as applying a differencing filter with a zero at  $z = 1$ . Extending the above argument, we see that by repeatedly applying the differencing filter, we will kill the constant term, then the linear, then the quadratic, and so on.

## 2.B.2 Vectors and Matrices of Polynomials

Notions of vectors and matrices can be combined with polynomials and rational functions, called either *vector/matrix of polynomials* or *polynomial vector/matrix*. For simplicity, we introduce all concepts on  $2 \times 1$  vectors and  $2 \times 2$  matrices.

A *vector of polynomials*, or, *polynomial vector* is given by

$$v(t) = \begin{bmatrix} \sum_{n=0}^N a_n t^n \\ \sum_{n=0}^N b_n t^n \end{bmatrix} = \begin{bmatrix} p(t) \\ q(t) \end{bmatrix} = \sum_{n=0}^N v_n t^n, \quad (2.287)$$

where  $v_n$  are  $2 \times 1$  vectors of scalars.

Similarly, a *matrix of polynomials*, or, *polynomial matrix* is given by

$$H(t) = \begin{bmatrix} \sum_{n=0}^N a_n t^n & \sum_{n=0}^N b_n t^n \\ \sum_{n=0}^N c_n t^n & \sum_{n=0}^N d_n t^n \end{bmatrix} = \begin{bmatrix} p(t) & q(t) \\ r(t) & s(t) \end{bmatrix} = \sum_{n=0}^N H_n t^n, \quad (2.288)$$

where  $H_n$  are  $2 \times 2$  matrices of scalars. In both of the above expressions,  $N$  is the maximum degree of any of the entries.

Rank is more subtle for polynomial matrices than for ordinary ones. For example, if  $\lambda = 3$ ,

$$H(t) = \begin{bmatrix} a + bt & 3(a + bt) \\ c + dt & \lambda(c + dt) \end{bmatrix}$$

is rank deficient for every value of  $t$ . On the other hand, if  $\lambda \neq 3$ , then it is rank deficient only if  $t = -a/b$  or  $t = -c/d$ , leading to the notion of *normal rank*. The normal rank of  $H(t)$  is the largest of the orders of the minors that have a determinant not identically zero. In the above example, for  $\lambda = 3$ , the normal rank is 1, while for  $\lambda \neq 3$ , the normal rank is 2.

A square polynomial matrix of full normal rank has an inverse, computed similarly to a scalar matrix as in (1.205),

$$H^{-1} = \frac{\text{adj}(H)}{\det(H)}. \quad (2.289)$$

A polynomial matrix  $H(t)$  is called *unimodular* if  $|\det H(t)| = 1$  for all  $t$ . The product of two unimodular matrices is unimodular, and the inverse of a unimodular matrix is unimodular as well. A polynomial matrix is unimodular if and only if its inverse is a polynomial matrix. All these facts can be proven using properties of determinants.



EXAMPLE 2.42 (UNIMODULAR POLYNOMIAL MATRIX) The determinant of the polynomial matrix

$$H(t) = \begin{bmatrix} 1+t & 2+t \\ t & 1+t \end{bmatrix}$$

is  $\det H(t) = (1+t)2 - t(2+t) = 1$ ; it is thus unimodular. Its inverse is

$$H(t) = \begin{bmatrix} 1+t & -(2+t) \\ -t & 1+t \end{bmatrix},$$

also a unimodular polynomial matrix.

**Vectors and Matrices of Laurent Polynomials** Just as polynomials can be extended to Laurent polynomials, vector/matrix of polynomials can be extended to vector/matrix Laurent polynomials,

$$H(t) = \begin{bmatrix} \sum_{n=-N}^N a_n t^n & \sum_{n=-N}^N b_n t^n \\ \sum_{n=-N}^N c_n t^n & \sum_{n=-N}^N d_n t^n \end{bmatrix} = \sum_{n=-N}^N H_n t^n,$$

and similarly for vectors. The normal rank is defined as for polynomial matrices. A polynomial matrix  $H(t)$  is called *Laurent unimodular* if  $|\det H(t)| = ct^k$  for all  $t \in \mathbb{Z}$ , some  $c \in \mathbb{C}$  and  $k \in \mathbb{Z}$ . The inverse of a Laurent polynomial matrix is again Laurent polynomial only if it is Laurent unimodular, since the adjugate in (2.289) is again a Laurent polynomial matrix, while the determinant is a monomial.

EXAMPLE 2.43 (LAURENT UNIMODULAR POLYNOMIAL MATRIX) The determinant of the Laurent polynomial matrix

$$H(t) = \frac{1}{4t} \begin{bmatrix} 1+3t & 1-3t \\ 3+t & 3-t \end{bmatrix}$$

is  $t^{-1}$ ; it is thus unimodular. Its inverse is

$$H(t)^{-1} = \frac{1}{4} \begin{bmatrix} 3-t & -(1-3t) \\ -(3+t) & 1+3t \end{bmatrix},$$

also a Laurent unimodular polynomial matrix.

**Vectors and Matrices of Ratios of Polynomials** A *rational matrix*, or, *matrix of rational functions* has entries that are ratios of polynomials,

$$H(t) = \begin{bmatrix} \frac{p_{00}(t)}{q_{00}(t)} & \frac{p_{01}(t)}{q_{01}(t)} \\ \frac{p_{10}(t)}{q_{10}(t)} & \frac{p_{11}(t)}{q_{11}(t)} \end{bmatrix},$$

where  $p_{ij}(t)$ ,  $q_{ij}(t)$  are polynomials in  $t$ . The normal rank is defined as for polynomial matrices. The inverse of a rational matrix is again a rational matrix. In Chapter 7 we will see a connection between polynomial matrices and FIR discrete-time filters; and a connection between rational matrices and IIR filters.

**Adjoint of a Polynomial Vector or Matrix** We now discuss the adjoint of a vector or matrix of polynomials; extensions to matrices of Laurent polynomials and rational functions follow similarly. When defining the adjoint of a vector or matrix of polynomials, we must ensure they are positive semidefinite on the unit circle, that is, for  $|t| = 1$ .<sup>68</sup> This is because we are extending the idea of autocorrelation (2.142) to vectors and matrices.

For simplicity, we consider polynomials with real coefficients first. The adjoint of a  $2 \times 1$  vector of polynomials (2.287) is

$$v(t)^* = v(t^{-1})^T = [p(t^{-1}) \quad q(t^{-1})]. \quad (2.290)$$

The product

$$v(t)^* v(t) = [p(t^{-1}) \quad q(t^{-1})] \begin{bmatrix} p(t) \\ q(t) \end{bmatrix} = p(t^{-1})p(t) + q(t^{-1})q(t)$$

is positive semidefinite on the unit circle  $|t| = 1$  since it is the sum of two positive semidefinite functions on the unit circle. The same hold for matrices of polynomials: the adjoint of a  $2 \times 2$  matrix of polynomials (2.288) is

$$H(t)^* = H(t^{-1})^T = \begin{bmatrix} p(t^{-1}) & r(t^{-1}) \\ q(t^{-1}) & s(t^{-1}) \end{bmatrix}; \quad (2.291)$$

the product  $H(t)^* H(t)$  is a Laurent matrix of polynomials, and positive semidefinite on the unit circle.

An extension of the spectral factorization Corollary 2.14 states:

**THEOREM 2.20** Let  $A(t)$  be a Laurent matrix of polynomials. Then, it is positive semidefinite on the unit circle, that is,  $A(e^{j\omega}) \geq 0$ , if and only if

$$A(t) = H(t)H^*((t^*)^{-1}), \quad (2.292)$$

where  $H(t)$  is a matrix of polynomials.

Such a matrix is given in Example (2.36) and its factorization in (2.250).

When polynomial coefficients are complex, the adjoint of the matrix in (2.288) is defined as

$$H(t)^* = H_*(t^{-1}) = \begin{bmatrix} p_*(t^{-1}) & r_*((t^{-1})^{-1}) \\ q_*((t^{-1})^{-1}) & s_*((t^{-1})^{-1}) \end{bmatrix} = \sum_{n=0}^N H_n^* t^{-n}, \quad (2.293)$$

where subscript  $*$  means Hermitian transpose of the coefficient matrices but without conjugating  $t$ , as is shown on the right-hand side of the equation.

<sup>68</sup>Our use of polynomial matrices is typically with the  $z$ -transform, that is,  $H(t) = H(z^{-1})$ , and thus, the polynomial turns into the DTFT on the unit circle.

**Paraunitary Matrices** The extension of unitary matrices (1.218) to matrices of polynomials or rational functions are *paraunitary* matrices. A square matrix of polynomials or rational functions  $U(t)$  is called paraunitary when it satisfies

$$U^*(t)U(t) = U_*(t^{-1})U(t) = I. \quad (2.294a)$$

Its inverse  $U^{-1}(t)$  equals its adjoint (Hermitian transpose)  $U^*(t) = U_*(t^{-1})$ . A real paraunitary matrix satisfies

$$U^T(t^{-1})U(t) = I. \quad (2.294b)$$

A paraunitary matrix is unitary on the unit circle, that is,

$$U^*(e^{j\omega})U(e^{j\omega}) = I. \quad (2.294c)$$

When all matrix entries are polynomials, the matrix is called *lossless*.

EXAMPLE 2.44 (PARAUNITARY MATRIX) The following matrix

$$U(t) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & t \end{bmatrix} \begin{bmatrix} \frac{1}{2} & -\frac{\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} & \frac{1}{2} \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1+\sqrt{3}t}{2} & \frac{-\sqrt{3}+t}{2} \\ \frac{1-\sqrt{3}t}{2} & \frac{-\sqrt{3}-t}{2} \end{bmatrix}$$

is paraunitary since (2.294b) is satisfied.

**Pseudocirculant Polynomial Matrices** The extension of circulant matrices (1.227) to polynomial matrices are *pseudocirculant* matrices, an example of which is (2.216). Such a matrix has polynomial entries and is circulant with entries above the diagonal multiplied by  $t$  (thus pseudocirculant), for example,

$$H(t) = \begin{bmatrix} h_0(t) & th_2(t) & th_1(t) \\ h_1(t) & h_0(t) & th_2(t) \\ h_2(t) & h_1(t) & h_0(t) \end{bmatrix}. \quad (2.295)$$

### 2.B.3 Kronecker Product

The Kronecker product of two matrices is defined as (we show a  $2 \times 2$  matrix as an example)

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \otimes M = \begin{bmatrix} aM & bM \\ cM & dM \end{bmatrix}, \quad (2.296)$$

where  $a, b, c$  and  $d$  are scalars and  $M$  is a matrix (neither matrix need be square). The Kronecker product has the following useful property with respect to the usual matrix product:

$$(A \otimes B)(C \otimes D) = (AC) \otimes (BD) \quad (2.297)$$

where all the matrix products have to be well-defined. See Exercise 2.8 for an application of Kronecker products.

## Chapter at a Glance

We now summarize the main concepts and results seen in this chapter, some in a tabular form. One of the key elements was finding the appropriate Fourier transform for a given space of sequences (such as  $\ell^2(\mathbb{Z})$  or finite-length ones circularly extended). That procedure can be summarized as:

- (i) Start with a given time shift  $\delta_{n-1}$ ;
- (ii) Induce an appropriate convolution operator  $Tx = Hx = h * x$ ;
- (iii) Find the eigensequences  $x_n$  of  $H$  ( $e^{j\omega n}$  for infinite-length sequences and  $e^{j(2\pi/N)kn}$  for finite-length sequences);
- (iv) Identify the frequency response as the eigenvalue corresponding to the above eigensequence ( $H(e^{j\omega n})$  for infinite-length sequences,  $H_k$  for finite-length sequences);
- (v) Find the appropriate Fourier transform by projecting the sequence on the spaces spanned by eigensequences identified in (iii) (discrete-time Fourier transform for infinite-length sequences, discrete Fourier transform for finite-length sequences).

Concept	Notation	Infinite-length sequences	Finite-length sequences
Shift	$\delta_{n-1}$	linear	circular
Sequence vector	$x_n$	$n \in \mathbb{Z}$	$n \in \{0, 1, \dots, N-1\}$
LSI system filter, impulse response operator	$h_n$	$n \in \mathbb{Z}$	$n \in \{0, 1, \dots, N-1\}$
Convolution	$h * x$	$\sum_{k \in \mathbb{Z}} x_k h_{n-k}$	$\sum_{k=0}^{N-1} x_k h_{N, (n-k) \bmod N}$
Eigensequence satisfies invariant space	$v$ $h * v_\lambda = \lambda v_\lambda$ $S_\lambda$	$e^{j\omega n}$ $h * v_\omega = H(e^{j\omega}) v_\omega$ $S_\omega = \{\alpha e^{j\omega n}\}$ $\alpha \in \mathbb{C}, \omega \in \mathbb{R}$	$e^{j(2\pi/N)kn}$ $h * v_k = H_k v_k$ $S_k = \{\alpha e^{j(2\pi/N)kn}\}$ $\alpha \in \mathbb{C}, k \in \mathbb{Z}$
Frequency response eigenvalue	$\lambda$	$\lambda_\omega = H(e^{j\omega})$ $\sum_{n \in \mathbb{Z}} h_n e^{-j\omega n}$	$\lambda_k = H_k$ $\sum_{n=0}^{N-1} h_n e^{-j(2\pi/N)kn}$
Fourier transform spectrum	$X$	DTFT $X(e^{j\omega}) = \sum_{n \in \mathbb{Z}} x_n e^{-j\omega n}$	DFT $X_k = \sum_{n=0}^{N-1} x_n e^{-j(2\pi/N)kn}$

**Table 2.11:** Concepts in discrete-time processing.

Concept	Expression
<b>Sampling factor 2</b>	
Input	$x_n, X(z), X(e^{j\omega})$
Downsampling by 2	$y_n = x_{2n}$ $y = D_2 x$ $Y(z) = (1/2) [X(z^{1/2}) + X(-z^{1/2})]$ $Y(e^{j\omega}) = (1/2) [X(e^{j\omega/2}) + X(e^{j(\omega-2\pi)/2})]$
Upsampling by 2	$y_n = \begin{cases} x_{n/2}, & n \text{ even;} \\ 0, & \text{otherwise.} \end{cases}$ $y = U_2 x$ $Y(z) = X(z^2)$ $Y(e^{j\omega}) = X(e^{j2\omega})$
Filtering by $h$ & downsampling by 2	$y = D_2 H x$ $Y(z) = (1/2) \frac{1}{2} [H(z^{1/2})X(z^{1/2}) + H(-z^{1/2})X(-z^{1/2})]$ $Y(e^{j\omega}) = (1/2) [H(e^{j\omega/2})X(e^{j\omega/2}) + H(e^{j(\omega-2\pi)/2})X(e^{j(\omega-2\pi)/2})]$
Upsampling by 2 & filtering by $g$	$y = G U_2 x$ $Y(z) = G(z)X(z^2)$ $Y(e^{j\omega}) = G(e^{j\omega})X(e^{j2\omega})$
<b>Sampling factor <math>N</math></b>	
Input	$x_n, X(z), X(e^{j\omega})$
Downsampling by $N$	$y_n = x_{Nn}$ $y = D_N x$ $Y(z) = (1/N) \sum_{k=0}^{N-1} X(\omega_N^k z^{1/N})$ $Y(e^{j\omega}) = (1/N) \sum_{k=0}^{N-1} X(e^{j(\omega-2\pi k/N)})$
Upsampling by $N$	$y_n = \begin{cases} x_{n/N}, & n = lN; \\ 0, & \text{otherwise.} \end{cases}$ $y = U_N x$ $Y(z) = X(z^N)$ $Y(e^{j\omega}) = X(e^{jN\omega})$

**Table 2.12:** Concepts in multirate discrete-time processing.

Domain	Autocorrelation/Crosscorrelation Properties		
<b>Sequences</b>			
		$x_n, y_n$	
Time	$a_n$	$\sum_{k \in \mathbb{Z}} x_k x_{k-n}^*$	$a_n = a_{-n}^*$
	$c_n$	$\sum_{k \in \mathbb{Z}} x_k y_{k-n}^*$	$c_{x,y,n} = c_{y,x,-n}^*$
DTFT	$A(e^{j\omega})$	$ X(e^{j\omega}) ^2$	$A(e^{j\omega}) = A^*(e^{j\omega})$
	$C(e^{j\omega})$	$X(e^{j\omega}) Y^*(e^{j\omega})$	$C_{x,y}(e^{j\omega}) = C_{y,x}^*(e^{j\omega})$
z-transform	$A(z)$	$X(z) X_*(z^{-1})$	$A(z) = A_*(z^{-1})$
	$C(z)$	$X(z) Y_*(z^{-1})$	$C_{x,y}(z) = C_{y,x*}(z^{-1})$
DFT	$A_k$	$ X_k ^2$	$A_k = A_{-k \bmod N}^*$
	$C_k$	$X_k Y_k^*$	$C_k = C_{y,x,-k \bmod N}^*$
<hr/>			
<b>Real sequences</b>		$x_n, y_n$	
Time	$a_n$	$\sum_{k \in \mathbb{Z}} x_k x_{k-n}$	$a_n = a_{-n}$
	$c_n$	$\sum_{k \in \mathbb{Z}} x_k y_{k-n}$	$c_{x,y,n} = c_{y,x,-n}$
DTFT	$A(e^{j\omega})$	$ X(e^{j\omega}) ^2$	$A(e^{j\omega}) = A(e^{-j\omega})$
	$C(e^{j\omega})$	$X(e^{j\omega}) Y(e^{-j\omega})$	$C_{x,y}(e^{j\omega}) = C_{y,x}(e^{-j\omega})$
z-transform	$A(z)$	$X(z) X(z^{-1})$	$A(z) = A(z^{-1})$
	$C(z)$	$X(z) Y(z^{-1})$	$C_{x,y}(z) = C_{y,x}(z^{-1})$
DFT	$A_k$	$ X_k ^2$	$A_k = A_{-k \bmod N}$
	$C_k$	$X_k Y_k$	$C_k = C_{y,x,-k \bmod N}$
<hr/>			
<b>Vector of sequences</b>		$\begin{bmatrix} x_{0,n} & x_{1,n} \end{bmatrix}^T$	
Time	$A_n$	$\begin{bmatrix} a_{0,n} & c_{0,1,n} \\ c_{1,0,n} & a_{1,n} \end{bmatrix}$	$A_n = A_{-n}^*$ $A_n = A_{-n}^T$
DTFT	$A(e^{j\omega})$	$\begin{bmatrix} A_0(e^{j\omega}) & C_{0,1}(e^{j\omega}) \\ C_{1,0}(e^{j\omega}) & A_1(e^{j\omega}) \end{bmatrix}$	$A(e^{j\omega}) = A^*(e^{j\omega})$ $A(e^{j\omega}) = A^T(e^{-j\omega})$
z-transform	$A(z)$	$\begin{bmatrix} A_0(z) & C_{0,1}(z) \\ C_{1,0}(z) & A_1(z) \end{bmatrix}$	$A(z) = A_*(z^{-1})$ $A(z) = A^T(z^{-1})$
DFT	$A_k$	$\begin{bmatrix} A_{0,k} & C_{0,1,k} \\ C_{1,0,k} & A_{1,k} \end{bmatrix}$	$A_k = A_{-k \bmod N}^*$ $A_k = A_{-k \bmod N}^T$

**Table 2.13:** Summary of concepts related to deterministic autocorrelation and cross-correlation of a sequence (upper half) and a vector of sequences (lower half). For vectors of sequences, an example for a vector of two sequences is given, and the second entry in properties is for real sequences. Note how we overload  $A(e^{j\omega})$ ,  $A(z)$  and  $A_k$  to mean both a scalar and a matrix depending on the sequence.

## Historical Remarks

**Cooley-Tukey FFT** The impact signal processing has had in practical terms is perhaps due in large part to the advent of the fast Fourier transform algorithms, spurred by the paper of Cooley and Tukey in 1965 [32]. It breaks the computation of the discrete Fourier transform of length  $N = N_1 N_2$  into a recursive computation of smaller DFTs of lengths  $N_1$  and  $N_2$ , respectively (see Section 2.9.1). Unbeknownst to them, a similar algorithm was published by Gauss some 150 years earlier, in his attempt to track asteroid trajectories.

**MP3, JPEG and MPEG** These already household names, are all algorithms for coding and compressing various types of data (audio, images and video). The basic ideas in all three all stem from transform-based work in signal processing, called subband coding, closely-related to wavelets as we will see in later chapters.

## Further Reading

**Books and Textbooks** A standard book on discrete-time processing is the one of Oppenheim and Schaffer [108], while a more recent one by Prandoni and one of the co-authors of this book, Vetterli, is a view of signal processing as needed for communications [115]. For statistical signal processing, see the text by Porat [113]. An early account of multirate signal processing was given by Crochiere and Rabiner [34]; a more recent one is by Vaidyanathan [158]. Dudgeon and Mersereau cover multidimensional signal processing in [48]. Blahut in [12] discusses fast algorithms for discrete-time signal processing, and in particular, various classes of FFTs.

**Inverse  $z$ -Transform Via Contour Integration** The formal inversion process for the  $z$ -transform is given by contour integration using Cauchy's integral formula when  $X(z)$  is a rational function of  $z$ . When  $X(z)$  is not rational, inversion can be quite difficult. A short account of inversion using contour integration is given in [89]; more details can be found in [108].

**Filter Design** Numerous filter design techniques exist. They all try to approximate the desired specifications of the system/filter by a realizable discrete-time system/filter. For IIR filters, one of the standard methods is to design the discrete-time filters from continuous-time ones using bilinear transformation. For FIR filters, windowing is often used to approximate the desired response by truncating it with a window, a topic we touched upon in Example 2.4. Then, linear phase is often incorporated as a design requirement. Kaiser window design method is a standard method for FIR filter design using windowing. Another commonly used design method is called Parks-McClellan. An excellent overview of filter design techniques is given in [108].

**Algebraic Theory of Signal Processing** Algebraic theory of signal processing is a recent development whose foundations can be found in [116]. It provides the framework for signal processing in algebraic terms, and allows for the development of other models but those based on time. For example, by introducing extensions different from the circular one we discussed in Section 2.6, in particular the symmetric one, one can show that the appropriate transform is the well-known DCT. Moreover, it provides a recipe for building a signal model bases on sequence and filter spaces, appropriate convolutions and appropriate Fourier transforms. As such, the existence and form of fast algorithms for computing such

transforms (among which are the well-known trigonometric ones) follow automatically. Some of the observations in this chapter were inspired by this algebraic framework.

**Pseudocirculant Matrices** We have seen the importance these matrices play in multi-rate systems, in particular in representing convolution in polyphase domain. A thorough presentation of such matrices can be found in [161].

**Stochastic Multirate Systems** In Section 2.8.4, we examined only the most basic of operations in stochastic multirate systems. A thorough discussion of various other special cases, including different periods for cyclostationarity of the input and the output, can be found in [126].

---

## Exercises with Solutions

### 2.1. Properties of Periodic Sequences

Consider the complex exponential sequences of the form

$$x_n = e^{j\omega_0 n}.$$

- (i) Show that if  $\alpha = \omega_0/2\pi$  is a rational number,  $\alpha = p/q$ , with  $p$  and  $q$  coprime integers, then  $x_n$  is periodic with period  $q$ .
- (ii) Show that if  $\alpha = \omega_0/2\pi$  is irrational, then  $x_n$  is not periodic.
- (iii) Show that if  $x$  and  $y$  are two periodic sequences with periods  $N$  and  $M$  respectively, then  $x + y$  is periodic with period  $\text{lcm}(M, N)$ .

*Solution:*

- (i) If  $\alpha = \frac{p}{q} = \frac{\omega_0}{2\pi}$  with  $p, q \in \mathbb{Z}$ , and  $(p, q) = 1$ , then  $\omega_0 = 2\pi\alpha = \frac{2\pi p}{q}$ , and

$$x_{n+q} = e^{j(n+q)\frac{2\pi p}{q}} = e^{jn\frac{2\pi p}{q}} e^{j2\pi p} = e^{jn\frac{2\pi p}{q}} = x_n,$$

and thus,  $x_n$  is periodic with period  $q$ .

- (ii) Suppose there exists  $N \in \mathbb{Z}$ , such that  $x_n = x_{n+N}$ . Then

$$e^{jn2\pi\alpha} = e^{j(n+N)2\pi\alpha} \Rightarrow 1 = e^{jN2\pi\alpha} \Rightarrow N\alpha = M \in \mathbb{Z} \Rightarrow \alpha = \frac{M}{N} \in \mathbb{Q},$$

which contradicts the assumption that  $\alpha$  is irrational.

- (iii) If  $x_n = x_{n+N}$  and  $y_n = y_{n+M}$ , then clearly  $z_n = z_{n+P}$ , where  $z_n = x_n + y_n$  and  $P = \text{lcm}(M, N)$ , since  $x_n = x_{n+P}$  and  $y_n = y_{n+P}$ .

Let us look into an example to show that the sequence  $z_n$  cannot have a smaller period. Consider  $x_n = n \bmod N$  and  $y_n = jn \bmod M$ . If  $z_n = x_n + y_n$  is periodic with period  $P$ , then

$$\begin{aligned} (n \bmod N) + j(n \bmod M) &= (n + P \bmod N) + j(n + P \bmod M) \\ (n \bmod N) &= (n + P \bmod N) \quad \text{and} \quad (n \bmod M) = (n + P \bmod M) \\ P \bmod N &= 0 \quad \text{and} \quad P \bmod M = 0 \\ P &= \text{lcm}(M, N) K \end{aligned}$$

for some  $K \in \mathbb{N}$ . Hence, the smallest possible period for  $z_n$  is  $\text{lcm}(M, N)$ .

### 2.2. LSI System Acting on Signals in $\ell^p(\mathbb{Z})$

Prove that if  $x \in \ell^p(\mathbb{Z})$  and  $h \in \ell^1(\mathbb{Z})$ , the result of  $h * x$  is in  $\ell^p(\mathbb{Z})$  as well.

*Solution:* [This solution is not correct.] By inclusion property (1.37),

$$1 < p \quad \Rightarrow \quad \ell^1(\mathbb{Z}) \subset \ell^p(\mathbb{Z}) \quad \Rightarrow \quad h \in \ell^p(\mathbb{Z}).$$



Then we use the definition of the  $\ell^p$  norm, (1.36a), to show

$$\begin{aligned} \|y\|_p^p &= \sum_{n \in \mathbb{Z}} |y_n|^p \stackrel{(a)}{=} \sum_{n \in \mathbb{Z}} \left| \sum_{k \in \mathbb{Z}} x_k h_{n-k} \right|^p \stackrel{(b)}{\leq} \sum_{n \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} |x_k|^p |h_{n-k}|^p \\ &\stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} \sum_{n \in \mathbb{Z}} |x_k|^p |h_{n-k}|^p \stackrel{(d)}{=} \sum_{k \in \mathbb{Z}} |x_k|^p \sum_{n \in \mathbb{Z}} |h_{n-k}|^p \stackrel{(e)}{\leq} M \sum_{k \in \mathbb{Z}} |x_k|^p \stackrel{(f)}{\leq} N, \end{aligned}$$

where (a) follows from the definition of convolution, (2.59); (b) from the triangle inequality (Definition 1.9(iii)); in (c) we exchanged summations as per (1.194); in (d) we pulled  $|x_k|^p$  in front of the sum over  $n$ ; (e) from  $h \in \ell^p(\mathbb{Z})$ ; and (f) from  $x \in \ell^p(\mathbb{Z})$ .

### 2.3. Filtering as a Projection

Given is a filter with impulse response  $g_n$ ,  $n \in \mathbb{Z}$ . For the filters below, check whether they are orthogonal projections or not:

(Hint: Use the frequency response  $G(e^{j\omega})$  of the filter's impulse response.)

- (i)  $g_n = \delta_{n-k}$ ,  $k \in \mathbb{Z}$ ;
- (ii)  $g_n = \frac{1}{2}\delta_{n+1} + \delta_n + \frac{1}{2}\delta_{n-1}$ ;
- (iii)  $g_n = \frac{1}{\sqrt{2}} \text{sinc}(\pi n/2)$ .

Consider now the class of real-valued discrete-time filters that perform an orthogonal projection. Give a precise characterization of this class (be more specific than just repeating the conditions for an operator to be an orthogonal projection).

*Solution:* We can represent our system as  $y = Gx$ , with the corresponding operator/matrix notation as in (2.63). To check that a filter (operator) is an orthogonal projection, we must check that it is idempotent and self-adjoint as in Definition 1.27.

Checking idempotency is easier in the Fourier domain since  $G^2 = G$  has as its frequency response pair  $G^2(e^{j\omega}) = G(e^{j\omega})$ . Checking self-adjointness is equivalent to checking that the operator matrix  $G$  is Hermitian.

- (i) We have that  $G(e^{j\omega}) = e^{-j\omega k}$  and  $G^2(e^{j\omega}) = e^{-j\omega 2k}$ . Thus,  $G^2(e^{j\omega}) \neq G(e^{j\omega})$ , unless  $k = 0$ ; this filter is not a projection operator.
- (ii) Similarly, we have

$$\begin{aligned} G(e^{j\omega}) &= \frac{1}{2}e^{j\omega} + 1 + \frac{1}{2}e^{-j\omega}, \\ G^2(e^{j\omega}) &= \left( \frac{1}{2}e^{j\omega} + 1 + \frac{1}{2}e^{-j\omega} \right)^2 = \frac{1}{4}e^{j2\omega} + e^{j\omega} + \frac{3}{2} + e^{-j\omega} + \frac{1}{4}e^{-j2\omega}. \end{aligned}$$

Thus,  $G^2(e^{j\omega}) \neq G(e^{j\omega})$ ; this filter is not a projection operator either.

- (iii) From Table 2.4, we can rewrite  $g_n$  as

$$g_n = \frac{1}{\sqrt{2}} \text{sinc}(\pi n/2) = \frac{1}{2\pi} \int_{-\pi/2}^{\pi/2} \sqrt{2} e^{j\omega n} d\omega,$$

yielding

$$G(e^{j\omega}) = \begin{cases} \sqrt{2}, & |\omega| \leq \pi/2; \\ 0, & \text{otherwise,} \end{cases} \quad G^2(e^{j\omega}) = \begin{cases} 2, & |\omega| \leq \pi/2; \\ 0, & \text{otherwise.} \end{cases}$$

Again,  $G^2(e^{j\omega}) \neq G(e^{j\omega})$ ; this filter is not a projection operator either. Note that for  $g_n = \text{sinc}(\pi n/2)$  we would have had a projection.

Trying to satisfy idempotency in the most general case, we see that for  $G^2(e^{j\omega})$  to be equal to  $G(e^{j\omega})$ ,  $G(e^{j\omega})$  can only be 1 or 0:

$$G(e^{j\omega}) = \begin{cases} 1, & \text{for } \omega \in R_1 \cup R_2 \cup \dots \cup R_n; \\ 0, & \text{otherwise,} \end{cases}$$

where  $R_1, R_2, \dots, R_n$  are disjoint regions in  $[-\pi, \pi)$ .

Self-adjointness is satisfied if  $g_n = g_{-n}^*$ , or, in Fourier domain,

$$G(e^{j\omega}) = \sum_n g_n e^{-j\omega n} = \sum_n g_{-n}^* e^{-j\omega n} = \sum_n g_n^* (e^{-j\omega n})^* = G^*(e^{j\omega}).$$

Since from the requirement for idempotency  $G(e^{j\omega})$  has to be real, self-adjointness is automatically satisfied.

#### 2.4. Fibonacci Filter

Given is a Fibonacci sequence:

$$\left[ \dots \quad 0 \quad \boxed{1} \quad 1 \quad 2 \quad 3 \quad 5 \quad 8 \quad 13 \quad \dots \right], \quad (\text{E2.4-1})$$

obtained via the following recursion:

$$y_n = y_{n-1} + y_{n-2}, \quad (\text{E2.4-2})$$

for  $n = 2, 3, \dots$ , and  $y_0 = y_1 = 1$ .

- (i) Create a filter whose impulse response  $h_n$  is given by the Fibonacci sequence. Is this filter FIR or IIR? Is it BIBO stable?
- (ii) Find the transfer function of the filter  $H(z)$ , draw the block-diagram of the system and show its pole-zero plot.
- (iii) Show that  $h_n$  is a sum of two geometric series:

$$h_n = a\alpha^n + b\beta^n.$$

*Solution:* The *Fibonacci filter* is one of the oldest known digital filters.

- (i) We may assume that the Fibonacci sequence in (E2.4-1) is the response of the system to the input  $x_n = \delta_n$ ,

$$h_n = \left[ \dots \quad 0 \quad \boxed{1} \quad 1 \quad 2 \quad 3 \quad 5 \quad 8 \quad 13 \quad \dots \right].$$

This is clearly an IIR filter as its impulse response is not finitely supported. Moreover, it is clearly not BIBO stable as a bounded input,  $\delta_n$ , creates an unbounded output,  $y_n = h_n$ .

- (ii) To find the transfer function we use the recursion (E2.4-2) to draw the block-diagram of the system as in Figure E2.4-1(a). From the diagram, it is easy to see that

$$Y(z) = \frac{1}{1 - z^{-1} - z^{-2}} X(z),$$

and thus

$$H(z) = \frac{1}{1 - z^{-1} - z^{-2}} = \frac{1}{(1 - \alpha z^{-1})(1 - \beta z^{-1})}.$$

with

$$\alpha = \frac{1 + \sqrt{5}}{2}, \quad \beta = \frac{1 - \sqrt{5}}{2}.$$

The constant  $\alpha$  is called the *golden ratio*. Its pole-zero plot is given in Figure E2.4-1(b).

- (iii) We use partial fraction expansion to get

$$H(z) = \frac{1}{1 - z^{-1} - z^{-2}} = \frac{a}{1 - \alpha z^{-1}} + \frac{b}{1 - \beta z^{-1}}, \quad \text{ROC} = \{z \mid |z| > \alpha\},$$

with

$$a = \frac{\alpha}{\sqrt{5}}, \quad b = -\frac{\beta}{\sqrt{5}}.$$

We can then use Table 2.6 to recognize the above as the  $z$ -transform of the following sequence:

$$\begin{aligned} h_n &= (a\alpha^n + b\beta^n)u_n = \frac{1}{\sqrt{5}} [\alpha^{n+1} - \beta^{n+1}] u_n \\ &= \frac{1}{\sqrt{5}} \left[ \left( \frac{1 + \sqrt{5}}{2} \right)^{n+1} - \left( \frac{1 - \sqrt{5}}{2} \right)^{n+1} \right]. \end{aligned}$$

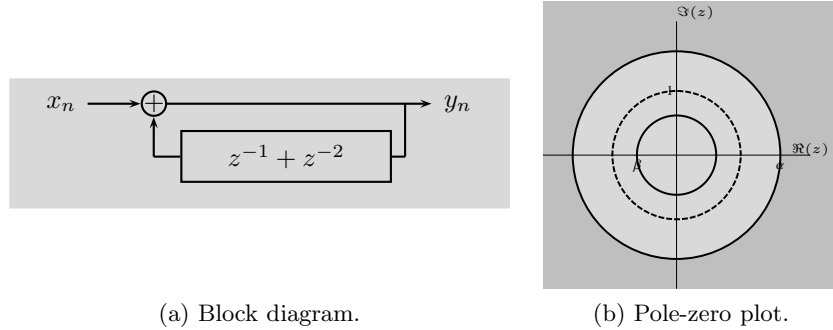


Figure E2.4-1: Fibonacci filter.

## 2.5. Circulant Matrices

Given is an  $N \times N$  circulant matrix  $C$  as in (1.227).

- (i) Give a formula for  $\det(C)$ .
- (ii) Give a simple test for the singularity of  $C$ .
- (iii) Prove that the eigenvalues of  $C$  are given by the frequency response (2.176a), the right eigenvectors are the columns of  $F$  and the left eigenvectors are the rows of  $F^*/N$ .
- (iv) Prove that  $C^{-1}$  is circulant.
- (v) Given two circulant matrices  $C_1$  and  $C_2$ , show that they commute,  $C_1 C_2 = C_2 C_1$ , and that the result is circulant as well.

*Solution:* The solution to this problem is based on the fact that the DFT diagonalizes the circulant convolution operator as in (2.177). Call  $C_k$  the DFT coefficients of  $c_0, c_1, \dots, c_{N-1}$  and  $\Lambda = \text{diag}([C_k])_{k=0}^{N-1}$ . The results now follow easily.

- (i) As the determinant of a product is equal to the product of determinants, from (2.177),

$$\begin{aligned} \det(C) &= \det(F \Lambda F^{-1}) = \det(F) \det(\Lambda) \det(F^{-1}) \\ &= \underbrace{\det(F) \det(F^{-1})}_{=1} \det(\Lambda) = \prod_{k=0}^{N-1} C_k. \end{aligned}$$

- (ii) From Part (i),  $C$  is nonsingular if and only if none of the  $C_k$  is zero.
- (iii) This follows directly from (2.177). In the spectral decomposition (1.210a), the eigenvalues of  $C$  are the elements of  $\Lambda$ ; we said that  $\Lambda = \text{diag}([C_k])_{k=0}^{N-1}$  where  $C_k$  are the DFT coefficients (frequency response) of the first column of  $C$ .

To show the eigenvector properties, write (2.177) as

$$\begin{aligned} C F &= F \Lambda, \\ C [v_0 \ v_1 \ \dots \ v_{N-1}] &= [v_0 \ v_1 \ \dots \ v_{N-1}] \text{diag}([C_k])_{k=0}^{N-1} \\ &= [C_0 v_0 \ C_1 v_1 \ \dots \ C_{N-1} v_{N-1}], \\ C v_k &= C_k v_k, \quad k = 0, 1, \dots, N-1, \end{aligned}$$

implying that the columns of  $F$  are the right eigenvectors of  $C$ .

The argument follows similarly for left eigenvectors,

$$F^{-1}C = \frac{1}{N}F^*C = \Lambda \frac{1}{N}F^*,$$

$$\frac{1}{N} \begin{bmatrix} v_0^* \\ v_1^* \\ \vdots \\ v_{N-1}^* \end{bmatrix} C = \text{diag}([C_k])_{k=0}^{N-1} \frac{1}{N} \begin{bmatrix} v_0^* \\ v_1^* \\ \vdots \\ v_{N-1}^* \end{bmatrix} = \begin{bmatrix} \frac{1}{N}C_0v_0^* \\ \frac{1}{N}C_1v_1^* \\ \dots \\ \frac{1}{N}C_{N-1}v_{N-1}^* \end{bmatrix},$$

$$v_k^*C = \frac{1}{N}C_kv_k^*, \quad k = 0, 1, \dots, N-1,$$

implying that the rows of  $F^*/N$  are the left eigenvectors of  $C$ .

(iv) Write  $C^{-1}$  as

$$C^{-1} = (F\Lambda F^{-1})^{-1} = F\Lambda^{-1}F^{-1}.$$

Since  $C^{-1}$  satisfies an equation of the same form as (2.177), it is circulant as well.

(v) This part follows similarly to the previous,

$$C_1C_2 = F\Lambda_1 \underbrace{F^{-1}F}_{=I} \Lambda_2 F^{-1} = F(\Lambda_1\Lambda_2)F^{-1} = F\Lambda F^{-1},$$

again satisfying an equation of the same form as (2.177);  $C_1C_2$  is thus circulant as well. Because  $\Lambda_1, \Lambda_2$  are diagonal matrices, they commute, allowing us to reverse the process and show that  $C_1$  and  $C_2$  commute.

## 2.6. Smoothing Operators

Given is

$$g_n = -(1/16)\delta_{n+1} + (1/4)\delta_n - (1/16)\delta_{n-1},$$

and the sequence of operations: filtering by  $\tilde{g}$ , followed by downsampling by 2, followed by upsampling by 2, followed by filtering by  $g$ , that is,  $y = GU_2D_2\tilde{G}x = Px$ .

- (i) Is  $g_n$  orthogonal to its even translates? Why?
- (ii) If the answer to Part (i) is *yes*: If  $\tilde{G} = G^T$ , is  $P$  an orthogonal projection? Why?

If the answer to Part (i) is *no*: Assume you are given  $\tilde{g}_n$  such that  $\{g_{n-2k}\}_{k \in \mathbb{Z}}$  and  $\{\tilde{g}_{n-2k}\}_{k \in \mathbb{Z}}$  are a pair of biorthogonal bases, where

$$\tilde{g}_n = \delta_{n+2} + a\delta_{n+1} + b\delta_n + a\delta_{n-1} + \delta_{n-2}.$$

How is the biorthogonality condition expressed in terms of their associated operators  $G$  and  $\tilde{G}$ ? Find constants  $a$  and  $b$ . Is  $P$  an orthogonal projection? Why?

*Solution:*

- (i) Take  $g_n$  and  $g_{n-2}$ . They overlap at  $n = 1$ , so  $\langle g_n, g_{n-2} \rangle = 1$ , and thus,  $g_n$  is not orthogonal to its even translates.
- (ii) Since the answer to the first question is *no*, we now assume that we are given  $\tilde{g}_n$  such that  $\{g_{n-2k}\}_{k \in \mathbb{Z}}$  and  $\{\tilde{g}_{n-2k}\}_{k \in \mathbb{Z}}$  are a pair of biorthogonal bases, implying that

$$\langle \tilde{g}_n, g_{n-2k} \rangle = \delta_k, \quad (\text{E2.6-1a})$$

$$\tilde{G}G = I. \quad (\text{E2.6-1b})$$

To find  $a$  and  $b$ , we use (E2.6-1a),

$$\begin{aligned} k=0 & \quad -\frac{1}{16}a + \frac{1}{4}b - \frac{1}{16}a = 1, \\ k=1 & \quad -\frac{1}{16}a + \frac{1}{4} = 0, \end{aligned}$$

yielding  $a = 4$  and  $b = 6$ , and giving rise to the second lowpass filter  $\tilde{g}_n$ .

We now check whether  $P = GU_2 D_2 \tilde{G}$  is an orthogonal projection.

$$\begin{aligned} P^2 &= (GU_2 D_2 \tilde{G})(GU_2 D_2 \tilde{G}) = GU_2 I D_2 \tilde{G} = P, \\ P^T &= (GU_2 D_2 \tilde{G})^T = \tilde{G}^T (U_2 D_2)^T G^T \neq P; \end{aligned}$$

it is a projection but not orthogonal.

### 2.7. Wiener Filtering

Consider Example 2.34 and squared error

$$\mathbb{E}[(x_n - \hat{x}_n)^2],$$

with  $\hat{x} = h * x + w$ . Find the minimum of the quadratic error by differentiating with respect to each filter coefficient  $h_n$  and verifying that you get the solution (2.241d) as in the example.

*Solution:* TBD.

### 2.8. Walsh Basis

The Walsh matrix  $W_N$  of size  $2^N \times 2^N$  is given by

$$W_N = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes W_{N-1}, \quad W_0 = [1], \quad W_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},$$

where  $\otimes$  is the Kronecker product (2.296).

- (i) Give  $W_2$  and  $W_3$ .
- (ii) Show that  $W_N$  is unitary (within a scale factor you should indicate).
- (iii) Create a block matrix  $T$

$$T = \begin{bmatrix} W_0 & & & \\ & \frac{1}{2^{1/2}} W_1 & & \\ & & \frac{1}{2} W_2 & \\ & & & \frac{1}{2^{3/2}} W_3 \\ & & & & \ddots \end{bmatrix},$$

and show that  $T$  is unitary. Sketch the upper left corner of  $T$ .

- (iv) The Walsh-Hadamard transform of size  $N$  is described via  $W_N$ . Derive an algorithm that uses  $N \log_2 N$  additions for a length- $N$  transform.

*Solution:*

- (i)

$$\begin{aligned} W_2 &= \begin{bmatrix} W_1 & W_1 \\ W_1 & -W_1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}, \\ W_3 &= \begin{bmatrix} W_2 & W_2 \\ W_2 & -W_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}, \end{aligned}$$

(ii) Let  $I_n$  denote the identity matrix of dimension  $n$ , and also note that  $W_N = W_N^T$ .

$$\begin{aligned} W_0^T W_0 &= 1 = I_1, \\ W_1^T W_1 &= \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = 2I_2, \\ W_2^T W_2 &= \begin{bmatrix} W_1 & W_1 \\ W_1 & -W_1 \end{bmatrix} \begin{bmatrix} W_1 & W_1 \\ W_1 & -W_1 \end{bmatrix} = 4I_4, \\ &\vdots \\ W_N^T W_N &= \begin{bmatrix} W_{N-1} & W_{N-1} \\ W_{N-1} & -W_{N-1} \end{bmatrix} \begin{bmatrix} W_{N-1} & W_{N-1} \\ W_{N-1} & -W_{N-1} \end{bmatrix} = 2^N I_{2^N}. \end{aligned}$$

Hence,  $(1/\sqrt{2^N})W_N$  is a  $2^N \times 2^N$  unitary matrix, for  $N \in \mathbb{N}$ .

(iii)

$$T = \begin{bmatrix} W_0 & & & \\ & \frac{1}{\sqrt{2}} W_1 & & \\ & & \frac{1}{2} W_2 & \\ & & & \ddots \end{bmatrix} = \begin{bmatrix} 1 & & & & & \\ & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & & & \\ & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & & & \\ & & & \frac{1}{2} & -\frac{1}{2} & \\ & & & \frac{1}{2} & \frac{1}{2} & \\ & & & & -\frac{1}{2} & \frac{1}{2} \\ & & & & \frac{1}{2} & -\frac{1}{2} \\ & & & & & \ddots \end{bmatrix}.$$

Note that  $T^T = T$ .

$$T^T T = \begin{bmatrix} W_0^T W_0 & & & \\ & \frac{1}{2} W_1^T W_1 & & \\ & & \frac{1}{2^2} W_2^T W_2 & \\ & & & \ddots \end{bmatrix} = \begin{bmatrix} I_1 & & & \\ & I_2 & & \\ & & I_4 & \\ & & & \ddots \end{bmatrix},$$

and thus,  $T$  is a unitary matrix.

(iv) A size  $2N$  Walsh-Hadamard transform  $WHT_{2N}$  is computed by evaluating a matrix product

$$\begin{bmatrix} WHT_N & WHT_N \\ WHT_N & -WHT_N \end{bmatrix} \begin{bmatrix} X_N^{(1)} \\ X_N^{(2)} \end{bmatrix}.$$

It follows that the addition cost  $\nu_N$  of the  $WHT_N$  satisfies the recursion formula  $\nu_{2N} = 2\nu_N + 2N$ . Using the fact that  $\nu_2 = 2$ , we find  $\nu_{2N} = 2^N + (N-1)2^N = N2^N$ , and the required  $N \log_2 N$  cost for a length- $N$  transform follows.

## Exercises

### 2.1. Sinusoidal Sequence

Given is the following sinusoidal sequence:

$$x_n = \sin \frac{\pi}{8} n, \quad y_n = x_n (u_n - u_{n-N}), \quad (\text{P2.1-1})$$

where  $u_n$  is the Heaviside sequence.

- (i) For  $N = 8, 12$  and  $16$ , sketch  $y_n$ .
- (ii) For which of the above three values of  $N$  is the following true? Sketch  $z_n$ .

$$z_n = \sum_{k \in \mathbb{Z}} y_{n-Nk} = x_n.$$

2.2. *Discrete Laplacian Operator*

Given is the following system:

$$y_n = x_{n-1} - 2x_n + x_{n+1}.$$

- (i) Is this system linear? Shift-invariant? Causal? Memoryless? BIBO stable?
- (ii) Does it have a matrix representation? If yes, write it down.
- (iii) For each of the following inputs, write and sketch the corresponding output of the system:

$$x_{0,n} = c; \quad x_{1,n} = \delta_n; \quad x_{2,n} = u_n,$$

where  $\delta_n$  is the Kronecker delta sequence, and  $u_n$  is the Heaviside sequence. Explain the effect of this system.

2.3. *Linear and Shift-Invariant Difference Equations*

Consider the difference equation (2.54).

- (i) Show, possibly using a simple example, that, if the initial conditions are nonzero, the system is not linear and not shift invariant.
- (ii) Show that, if the initial conditions are zero, then (a) the homogeneous solution is zero, (b) the system is linear and (c) the system is shift invariant.

2.4. *Geometric Sequences and Their Properties*

Given is a geometric sequence as in (2.6) with  $|\alpha| < 1$ ,

$$x_n = \begin{cases} 0, & \text{for } n < 0; \\ \gamma \alpha^n, & \text{for } n \geq 0. \end{cases}$$

- (i) Show that  $\gamma = \sqrt{1 - \alpha^2}$  leads to a unit-norm sequence, that is,  $\|x\|_2 = 1$ .
- (ii) Compute the autocorrelation of this unit-norm geometric sequence.
- (iii) Compute the convolution of this unit-norm geometric sequence with itself.
- (iv) Call  $y_n$  a different unit-norm geometric sequence. Compute the crosscorrelation between  $x_n$  and  $y_n$ , as well as their convolution.

2.5. *Deterministic Autocorrelation and Crosscorrelation*

Consider deterministic autocorrelation and crosscorrelation sequences and their DTFTs as in (2.96) and (2.99). Show that:

- (i)  $a_n = a_{-n}^*$ , (2.17a), and  $|a_n| \leq a_0$ ;
- (ii)  $c_n$  is in general not symmetric but rather Hermitian symmetric as in (2.20a), and  $C(e^{j\omega}) = X(e^{j\omega}) Y^*(e^{j\omega})$ ;
- (iii) The generalized Parseval's equality (2.104) holds.

2.6. *Modulation Property of the DTFT*

Given are two sequences  $x$  and  $h$ , both in  $\ell^1(\mathbb{Z})$ . Verify that the convolution in frequency property (2.94) holds:

$$h_n x_n \xrightarrow{\text{DTFT}} \frac{1}{2\pi} H *_{\omega} X.$$

2.7. *Thirdband Filter*

Given is the following filter:

$$h_n = \sqrt{3} \frac{\sin(\pi n/3)}{\pi n}.$$

- (i) Find the DTFT of  $h_n$ . What kind of a filter is it?
- (ii) Given  $x_n = (\delta_n + \delta_{n-1})/2$ , and  $y = h * x$ , sketch  $|Y(e^{j\omega})|$ .

2.8. *Bandlimitedness*

Are signals with all odd samples equal to zero bandlimited? Prove or disprove (by constructing a counter-example) the assertion and draw the spectrum of an example sequence demonstrating your point.

2.9. *ROC of z-Transform*

Compute the  $z$ -transforms and associated ROCs for the following sequences:

- (i)  $\delta_n$ .
- (ii)  $\delta_{n-k}$ .
- (iii)  $\alpha^n u_n$ .
- (iv)  $-\alpha^n u_{-n-1}$ .
- (v)  $n\alpha^n u_n$ .
- (vi)  $-n\alpha^n u_{-n-1}$ .
- (vii)  $\cos(\omega_0 n) u_n$ .
- (viii)  $\sin(\omega_0 n) u_n$ .
- (ix)  $\alpha^n$  for  $0 \leq n \leq N$ , and 0 otherwise.

2.10. *Regions of Convergence*

Consider rational  $z$ -transforms  $X(z)$  with  $ROC_X$  and  $Y(z)$  with  $ROC_Y$ . Find the ROCs of the following:

$$\begin{aligned} A(z) &= X(z) + Y(z); \\ B(z) &= X(z)Y(z). \end{aligned}$$

2.11. *Orthogonality*

Consider a sequence  $p_n$  with  $z$ -transform  $P(z)$ . The goal is to find sequences satisfying orthogonality constraint with respect to all shifts,

$$\langle p_n, p_{n-k} \rangle = \delta_k \iff P(z)P(z^{-1}) = 1.$$

- (i) If  $p_n$  is FIR, that is,  $P(z)$  is a polynomial, show that the only possible solution is

$$p_n = \pm \delta_{n-l}, \quad \text{with arbitrary } l \in \mathbb{Z}. \quad (\text{P2.11-1})$$

- (ii) If the sequence  $p_n$  has a rational  $z$ -transform  $P(z)$ , show that

$$P(z) = \frac{A(z)}{\tilde{A}(z)}, \quad (\text{P2.11-2})$$

where  $A(z)$  is a polynomial of degree  $(L-1)$  and  $\tilde{A}(z) = z^{-L+1}A(z^{-1})$ , will be a solution.

2.12. *Linear and Circular Convolution as Polynomial Products*

Given two polynomials of degree  $N-1$ ,

$$A(z) = \sum_{n=0}^{N-1} a_n z^n, \quad B(z) = \sum_{n=0}^{N-1} b_n z^n, \quad (\text{P2.12-1})$$

show that  $C(z) = A(z)B(z) \bmod (z^N - 1) = \sum_{n=0}^{N-1} c_n z^n$  is equivalent to the circular convolution of the sequences  $[a_0 \ a_1 \ \dots \ a_{N-1}]^T$  and  $[b_0 \ b_1 \ \dots \ b_{N-1}]^T$ , or

$$c_n = \sum_{k=0}^{N-1} a_{(n-k) \bmod N} b_k. \quad (\text{P2.12-2})$$

(Hint: The operation  $A(z)B(z) \bmod (z^N - 1)$  is the remainder of the division of  $A(z)B(z)$  by  $(z^N - 1)$ .)

2.13. *Deterministic Autocorrelation*

The deterministic autocorrelation for a real-valued stable sequence  $x_n$  is defined as in (2.16),

$$a_n = \sum_{k \in \mathbb{Z}} x_k x_{k-n}.$$

- (i) Show that the  $z$ -transform of  $a_n$  is  $A(z) = X(z)X(z^{-1})$ . Determine the region of convergence for  $A(z)$ .
- (ii) If  $x_n = \alpha^n u_n$ , with  $u_n$  the Heaviside sequence (2.10), show the pole-zero plot for  $A(z)$ , including the region of convergence. Find  $a_n$  by evaluating the inverse  $z$ -transform of  $A(z)$ .



- (iii) Specify another sequence,  $y_n$ , that is not equal to  $x_n$  from Part (ii), but has the same deterministic autocorrelation sequence as that of  $x_n$ .
- (iv) Specify a third sequence,  $v_n$ , that is not equal to either  $x_n$  or  $y_n$  but has the same deterministic autocorrelation sequence as that of  $x_n$ .

2.14. *Allpass Filters*

Consider allpass filters where

$$H(z) = \prod_{i=1}^N \frac{z^{-1} - a_i^*}{1 - a_i z^{-1}}.$$

- (i) Assume the filter has real coefficients. Show pole-zero locations, and that numerator and denominator polynomials are mirrors of each other, that is  $D(z) = z^{-N} N(z^{-1})$ .
- (ii) Given  $h_n$ , the causal, real-coefficient impulse response of a stable allpass filter, give its deterministic autocorrelation  $a_k = \sum_n h_n h_{n-k}$ . Show that the set  $\{h_{n-k}\}, k \in \mathbb{Z}$ , is an orthonormal basis for  $\ell^2(\mathbb{Z})$ .
- (iii) Show that the set  $\{h_{n-2k}\}$  is an orthonormal set but not a basis for  $\ell^2(\mathbb{Z})$ .

2.15. *Allpass System*

Given is an allpass system,

$$H(z) = C \prod_{k=1}^M \frac{z^{-1} - \alpha_k}{1 - \alpha_k z^{-1}}, \quad \alpha_k \in \mathbb{R}.$$

- (i) Find its magnitude on the unit circle, that is,  $|H(e^{j\omega})|$ . Specify the value of  $C$  for that magnitude to equal 1.
- (ii) Show that this filter will preserve the norm of any sequence filtered by it.

2.16. *Block Circulant Matrices*

A block-circulant matrix of size  $NM \times NM$  is like a circulant matrix of size  $N \times N$ , except that the elements are now blocks of size  $M \times M$ . For example, given two  $M \times M$  matrices  $A$  and  $B$ ,

$$C = \begin{bmatrix} A & B \\ B & A \end{bmatrix}$$

is a size  $2M \times 2M$  block-circulant matrix. Show that block-circulant matrices are block-diagonalized by block Fourier transforms of size  $NM \times NM$  defined as

$$F_{NM}^B = F_N \otimes I_M,$$

where  $F_N$  is an  $N \times N$  Fourier matrix,  $I_M$  is an  $M \times M$  identity matrix and  $\otimes$  is the Kronecker product (2.296).

2.17. *Pattern Recognition*

In pattern recognition, it is sometimes useful to expand a signal using the desired pattern (template) and its shifts, as basis functions. For simplicity, consider a signal of length  $N$ ,  $x_n, n = 0, \dots, N-1$ , and a pattern  $p_n, n = 0, \dots, N-1$ . Then, choose as basis functions

$$\varphi_{kn} = p_{(n-k) \bmod N}, \quad k = 0, \dots, N-1,$$

that is, circular shifts of  $p_n$ .

- (i) Derive a simple condition on  $p_n$  so that any  $x_n$  can be written as a linear combination of  $\{\varphi_k\}$ .
- (ii) Assuming the previous condition is met, give the coefficients  $\alpha_k$  of the expansion

$$x_n = \sum_{k=0}^{N-1} \alpha_k \varphi_{k,n}.$$

2.18. *Computing Linear Convolution with the DFT*

Prove that the linear convolution of two sequences of length  $M$  and  $L$  can be computed using DFTs of size  $N \geq M + L - 1$ , and show how to do it.

2.19. *DFT Properties*

Find the DFT pairs for the following:

- (i) Time-reversed sequence  $x_{-n \bmod N}$ ;
- (ii) Real symmetric sequence  $x_n = x_{-n \bmod N}$  and real antisymmetric sequence  $x_n = -x_{-n \bmod N}$ ;
- (iii) Convolution in frequency property (2.168).

2.20. *Downsampling by  $N$* Prove the  $z$ -transform and the DTFT transform pairs for downsampling by  $N$  given by (2.183) and (2.184), respectively.2.21. *Downsampling*Given is a length- $N$  sequence  $x_n$ . Let  $N = m_0 M$ , where  $m_0$  and  $M$  both are positive integers, and  $y_n$  be a sequence obtained by downsampling  $x_n$  by  $M$ , that is,

$$y_n = x_{Mn}, \quad n = 0, 1, \dots, m_0 - 1.$$

Let  $Y_k$  be the length- $(N/M)$  DFT of the sequence  $y_n$ , and  $X_k$  be the length- $N$  DFT of the sequence  $x_n$ . Prove the following:

- (i) For  $M = 2$ , the DFT of the downsampled sequence  $y_n$  is

$$Y_k = \frac{1}{2} (X_k + X_{k+N/2}) \quad k = 0, 1, \dots, \frac{N}{2} - 1.$$

- (ii) For arbitrary  $M$ , the DFT of the downsampled sequence  $y_n$  is

$$Y_k = \frac{1}{M} \sum_{i=0}^{M-1} X_{k+iN/M}, \quad k = 0, 1, \dots, \frac{N}{M} - 1.$$

2.22. *Multirate System with Different Sampling Rates*

Consider the system

$$y = HD_4GD_2FD_3x.$$

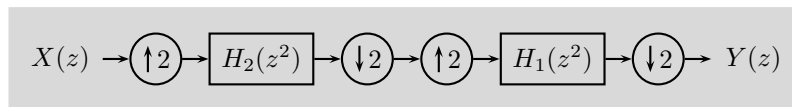
- (i) Draw a block diagram.
- (ii) Derive an equivalent system consisting of a single filter block and a single downsampling matrix. Write the  $z$ -transform of the equivalent system as a function of  $F(z)$ ,  $G(z)$  and  $H(z)$ .

2.23. *Interchange of Filtering and Sampling Rate Change*

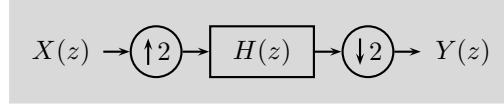
- (i) Prove that downsampling by 2 followed by filtering with  $\tilde{G}(z)$  is equivalent to filtering with  $\tilde{G}(z^2)$  followed by downsampling by 2.
- (ii) Prove that filtering with  $G(z)$  followed by upsampling by 2 is equivalent to upsampling by 2 followed by filtering with  $G(z^2)$ .

2.24. *Multirate Identities*

- (i) Find the overall transfer function  $Y(z)/X(z)$  of the system in Figure P2.24-1.

**Figure P2.24-1:** Multirate system 1.

- (ii) In the system in Figure P2.24-2, if  $H(z) = H_0(z^2) + z^{-1}H_1(z^2)$ , prove that  $Y(z) = X(z)H_0(z)$ .

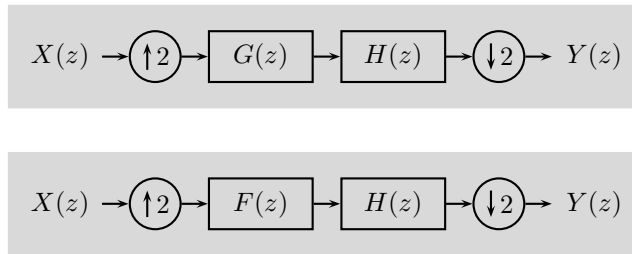
**Figure P2.24-2:** Multirate system 2.

(iii) Let  $H(z)$ ,  $F(z)$  and  $G(z)$  be filters satisfying

$$H(z)G(z) + H(-z)G(-z) = 2, \quad (\text{P2.24-1a})$$

$$H(z)F(z) + H(-z)F(-z) = 0. \quad (\text{P2.24-1b})$$

Prove that for one of the systems in Figure P2.24-3  $Y(z)/X(z) = 1$ , while for the other  $Y(z)/X(z) = 0$ .

**Figure P2.24-3:** Multirate system 3.

### 2.25. Interchange of Multirate Operations and Filtering

For the system given by the input-output relation

$$y = D_2 A D_2 A D_2 A x,$$

where  $A$  is a matrix representing a filter:

- (i) With the use of identities for the interchange of multirate operations and filtering, find the simplest equivalent system,  $y = D_n H x$ . Specify the downsampling factor  $n$  and write  $H$  in the  $z$ -transform and Fourier domains.
- (ii) If  $A$  is an ideal halfband lowpass filter, draw  $|H(e^{j\omega})|$ , clearly specifying the cut-off frequencies.
- (iii) If  $A$  is an ideal halfband highpass filter, draw  $|H(e^{j\omega})|$ , clearly specifying the cut-off frequencies. Is this transfer function capturing the highest frequency content in the sequence  $x$ ? Explain.

### 2.26. Commutativity of Up- and Downsampling

Prove that downsampling by  $M$  and upsampling by  $N$  commute if and only if  $M$  and  $N$  are coprime.

### 2.27. Combinations of Upsampling and Downsampling

Using matrix notation, compare:

- (i)  $U_3 D_2 x$  to  $D_2 U_3 x$ ;
- (ii)  $U_4 D_2 x$  to  $D_2 U_4 x$ .

Explain the outcome of these comparisons.

### 2.28. Periodically Shift-Varying Systems

Show that an LPSV system of period  $N$ , can be implemented with a polyphase transform followed by upsampling by  $N$ ,  $N$  filter operations and a summation.

2.29. *Convolution and Sum of Discrete Random Variables*

A random variable is *discrete* when it takes values in a countable set. A discrete random variable  $x$  has a *probability mass function* (PMF)  $p_x$  defined by  $p_x(k) = P(x = k)$ . Let  $x$  and  $y$  be independent, integer-valued random variables with PMFs  $p_x$  and  $p_y$ . Show that  $z = x + y$  has PMF  $p_z = p_x * p_y$ .

2.30. *Toeplitz Matrix-Vector Products*

Given a size- $(N \times N)$  Toeplitz matrix  $T$ , and a length- $N$  vector  $x$ , show that the product  $Tx$  can be computed with  $O(N \log_2 N)$  operations. The method consists in extending  $T$  into a circulant matrix  $C$ . What is the minimum size of  $C$ , and how does it change if  $T$  is symmetric?

## Chapter 3

# Functions and Continuous-Time Systems

## Contents

3.1	Introduction . . . . .	318
3.2	Functions . . . . .	319
3.3	Systems . . . . .	327
3.4	Fourier Transform . . . . .	334
3.5	Fourier Series . . . . .	355
3.6	Continuous Stochastic Processes and Systems . .	363
	Chapter at a Glance . . . . .	368
	Historical Remarks . . . . .	369
	Further Reading . . . . .	370
	Exercises with Solutions . . . . .	370
	Exercises . . . . .	372

As in Chapter 2, the key word in the title of this chapter is *time*; contrasting with Chapter 2, *discrete* is now replaced with *continuous*. Time is now uncountable rather than sampled as before. Our vectors are now *functions* (the domain is continuous time), and as we saw in Chapter 1 these form the vector space  $\mathbb{C}^{\mathbb{R}}$ . Restricting to the normed vector spaces  $\mathcal{L}^2(\mathbb{R})$  and  $\mathcal{L}^\infty(\mathbb{R})$  corresponds to the physical phenomena of finite energy and boundedness. Operators that map a function to a function are called *continuous-time systems*. A continuous-time system with the shift-invariance property is described by *convolution* with the system's *impulse response*. An impulse response is a function, and we will see that it is appropriate to require that it belong to  $\mathcal{L}^1(\mathbb{R})$  or  $\mathcal{L}^2(\mathbb{R})$ . Once the convolution operation is defined, spectral theory allows us to construct the *Fourier transform*.

As in Chapter 2, the above discussion implicitly assumed that the underlying domain, time, is infinite. In practice we glimpse a finite portion of time. In Chapter 2, we dealt with this issue by assuming that a finite sequence was circularly extended leading to the notion of *circular convolution* and the *discrete Fourier transform*; in this chapter, finitely-supported functions circularly extended (peri-

odized) will also have an appropriate circular convolution as well as an appropriate Fourier transform, the *Fourier series*.

### 3.1 Introduction

In most of Chapter 2, we considered sequences; here, we look at functions defined for all times  $t \in \mathbb{R}$ . Such a function,

$$x(t), \quad t \in \mathbb{R}, \quad (3.1)$$

could be the sound pressure sensed by a microphone, or the temperature at a sensor, etc; the key is that a value exists at every time.

In real life, we observe only a finite portion of a function on the real line,

$$x(t), \quad t \in [0, T]. \quad (3.2)$$

Moreover, computations are always done on finite inputs, requiring a decision on what happens at the boundaries. As with sequences, we typically extend functions on a finite interval circularly (periodizing the function in the process),

$$x(t + T) = x(t), \quad t \in \mathbb{R}. \quad (3.3)$$

While finite-length functions in (3.2) and infinite-length periodic ones in (3.3) have a fundamentally different character, we will use the same types of tools to analyze them. Techniques designed explicitly for finite-length functions are mathematically rooted in treating the function as one period of an infinite-length periodic function. The consequences of this implicit periodization are central in signal processing.

As in Chapter 2, we thus define two broad classes of functions for which to develop our tools:

- (i) *Functions on the real line* are the vector space  $\mathbb{C}^{\mathbb{R}}$  of functions with domain  $\mathbb{R}$ , as defined in (1.17c). The support of a function may be a proper subset of  $\mathbb{R}$ ; for example, we will often consider functions on the real line that are nonzero only at nonnegative time.
- (ii) *Functions on a finite interval*, without loss of generality, have support in  $[0, T)$ . The tools we will develop do not treat the vector space of functions with support in  $[0, T)$  generically, but rather as functions defined on a circular domain.

Functions we consider are typically bounded, often smooth, and sometimes periodic. An important space of functions are those that are bandlimited, that is, the maximum frequency present in the function is finite. This plays an important role in signal processing, since bandlimited functions can be recovered exactly from samples, making a natural connection to discrete-time sequences.

Given functions (signals, vectors), one can apply operators (systems, filters), for example, the voltage produced at a microphone in response to pressure variations. These map input functions into output ones, and since they involve continuous-time functions, they are usually called *continuous-time systems* (operators). Many

continuous-time systems, such as the microphone described above, are physical systems governed by differential equations: for example, the sound waves reaching a microphone obey the wave equation. Often, these differential equations have a smoothing effect, and functions with singularities (such as a point of discontinuity) are smoothed by the time they are observed. In the microphone example, a gunshot is first smoothed by the wave equation, and further smoothed by the microphone itself. We mention these effects to emphasize the difference between mathematical abstractions and observed phenomena. These differential equations are often linear, or even linear and shift-invariant; in Chapter 2, the same was true of difference equations.

### Chapter Outline

From this short introduction, the outline of the chapter follows naturally. Section 3.2 discusses continuous-time functions, where we introduce function spaces of interest and comment on local and global smoothness. We follow with a short overview of continuous-time systems in Section 3.3, particularly LSI systems stemming from linear constant-coefficient differential equations. This discussion leads to the convolution operator and its properties, such as stability. Section 3.4 reviews the Fourier transform and its properties. We emphasize the eigenfunction property of complex exponentials and give key relations of the Fourier transform, together with properties for certain function spaces. We briefly discuss the Laplace transform, an extension of the Fourier transform akin to the  $z$ -transform seen in the previous chapter, allowing us to deal with larger classes of functions. In Section 3.5, we discuss the natural orthonormal basis for periodic functions given by the Fourier series. We study circular convolution and the eigenfunction property of complex exponentials as well as the properties of the Fourier series. Then, we explore the duality with the DTFT for sequences seen in Chapter 2. In Section 3.6, we study continuous stochastic processes and systems.

## 3.2 Functions

### 3.2.1 Functions on the Real Line

The set of functions in (3.1), where  $x(t)$  is either real or complex, together with vector addition and scalar multiplication, forms a vector space (see Definition 1.1). The inner product between two functions on the real line is defined in (1.20c), and induces the standard  $\mathcal{L}^2$  (or Euclidean) norm (1.23c). Other norms of interest are the  $\mathcal{L}^1$  norm from (1.38a) with  $p = 1$ , and the  $\infty$  norm from (1.38b). We now look into a few spaces of interest.

### Function Spaces

**Space of Square-Integrable Functions  $\mathcal{L}^2(\mathbb{R})$**  The constraint of a finite square norm is necessary for turning the vector space  $\mathbb{C}^{\mathbb{R}}$  defined in (1.17c) into the Hilbert

space of *finite-energy* functions  $\mathcal{L}^2(\mathbb{R})$ . As for sequences, this space affords a geometric view, for example, the orthogonal projection theorem.

**Space of Bounded Functions  $\mathcal{L}^\infty(\mathbb{R})$**  The space of bounded functions contains all functions  $x(t)$  such that, for some finite  $M$ ,  $|x(t)| \leq M$  for all  $t \in \mathbb{R}$ . This space is denoted  $\mathcal{L}^\infty(\mathbb{R})$  since it consists of functions with finite  $\mathcal{L}^\infty$  norm.

**Space of Absolutely-Integrable Functions  $\mathcal{L}^1(\mathbb{R})$**  The space of absolutely-integrable functions consists of those with finite  $\mathcal{L}^1$  norm.

The  $\mathcal{L}^p$  spaces do not satisfy a nesting property as the  $\ell^p$  sequence spaces do in (1.37). Therefore, to avoid technical difficulties when inclusion in different  $\mathcal{L}^p$  spaces is needed, we restrict attention to functions in the intersection of the spaces. when inclusion in different  $\mathcal{L}^p$  spaces is needed to avoid technical difficulties, we restrict to the intersection of the spaces. For example, certain theorems apply only to functions that are both absolutely-integrable as well as square-integrable, that is, those belonging to  $\mathcal{L}^1 \cap \mathcal{L}^2$ .

**Spaces of Smooth Functions** To describe the global smoothness of a function, we use its continuity and continuity of its derivative(s); these form the  $C^q$  spaces we saw in Section 1.2.4. Even a single point where the  $q$ th derivative does not exist or is not continuous prevents membership in  $C^q$ . Thus, global smoothness can fail to capture distinctions between important, frequently-encountered types of functions, and those that are quite esoteric. For example, the simple function  $u(t) = 1$  for  $t \geq 0$  and  $u(t) = 0$  for  $t < 0$  is infinitely-differentiable at every nonzero  $t$ , but it fails to be even in  $C^0$ ; in global smoothness, it is no different than a function that is discontinuous everywhere. Therefore, to differentiate functions in terms of smoothness, we consider local smoothness as well.

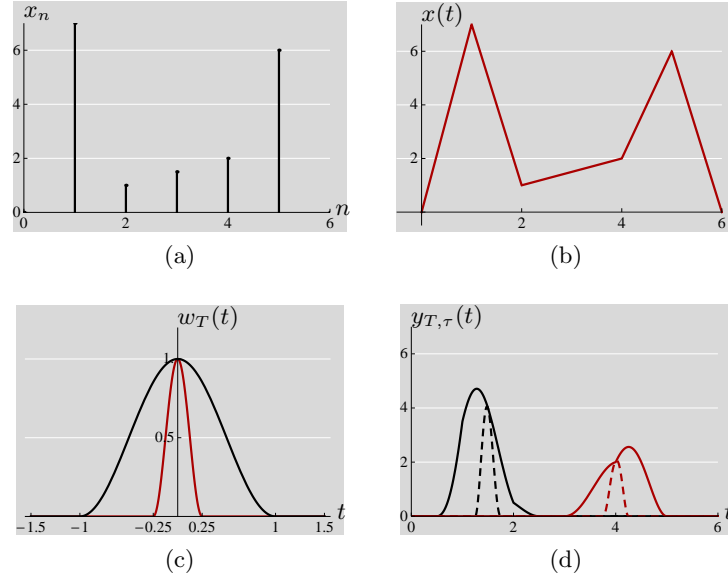
In calculus, it is natural to look at differentiability at various points in the domain of the function. In signal processing, on the other hand, it is often preferable to define local smoothness using the global smoothness of a windowed version of a function. We illustrate this with an example.

**EXAMPLE 3.1 (CONTINUOUS AND PIECEWISE LINEAR FUNCTION)** Let  $\{x_0, x_1, \dots, x_L\}$  be a finite sequence of real values with  $x_0 = x_L = 0$  as in Figure 3.1(a). Construct a continuous-time function

$$x(t) = \begin{cases} x_n + (t - n)(x_{n+1} - x_n), & \text{for } n \leq t < n + 1, n \in \{0, 1, \dots, L - 1\}; \\ 0, & \text{for } t \notin [0, L), \end{cases} \quad (3.4a)$$

a linear interpolation between the integer points as in Figure 3.1(b) such that  $x(n) = x_n$ . This function is in  $\mathcal{L}^1$ ,  $\mathcal{L}^2$ , and  $\mathcal{L}^\infty$ , since the sequence  $\{x_n\}$  is finite and bounded. In terms of smoothness, looking only at a single linear piece, the function seems to be in  $C^\infty$ , but since the function is not differentiable at the integers, it is only in  $C^0$ .





**Figure 3.1:** (a) A finite sequence of real values  $x_n = \{x_0, x_1, \dots, x_6\}$ . (b) The piecewise-linear function  $x(t)$  obtained by linearly interpolating  $x_n$  via (3.4a). (c) Two different windows  $w_T(t)$  from (3.4b) for  $T = 2$  and  $T = 1/2$ . (d) Four windowed versions  $y_{T,\tau}$  for  $T \in \{1/2, 2\}$  and  $\tau \in \{1/8, 3/8\}$ . All four are in  $C^0$ , and only  $y_{1/2,3/8}$  is in  $C^1$ .

We investigate local smoothness using a window that is in  $C^1$ , for example,

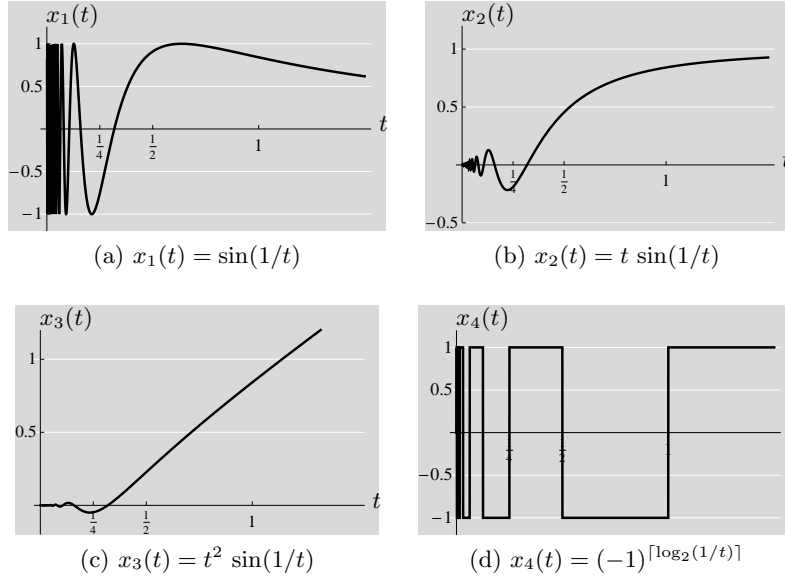
$$w_T(t) = \begin{cases} \frac{1}{2}(1 + \cos(2\pi t/T)), & \text{for } |t| \leq T/2; \\ 0, & \text{otherwise,} \end{cases} \quad (3.4b)$$

where  $T > 0$ . The window is of size  $T$  and centered around the origin. It has one continuous derivative, that is,  $w_T \in C^1$ . Figure 3.1(c) shows  $w_T$  for two values of  $T$ . We will use all shifts of  $w_T$ .

For any fixed width parameter  $T$  and shift parameter  $\tau$ , define the windowed version of  $x(t)$ :

$$y_{T,\tau}(t) = x(t) w_T(t - \tau). \quad (3.4c)$$

The global smoothness of  $y_{T,\tau}$  varies based on the parameters  $T$  and  $\tau$  and gives us a local smoothness of  $x$ . As a product of continuous functions,  $y_{T,\tau}$  is always continuous (that is, in  $C^0$ ). When  $T > 1$ , the support of the shifted window will always include at least one integer point—no matter what  $\tau$  is; thus,  $y_{T,\tau}$  will not be in  $C^1$ . When  $T < 1$ , depending on  $T$  and  $\tau$ , some of the windowed versions will be in  $C^1$ . For example, if  $T = 1/2$ , then for  $\tau \in [n + 1/4, n + 3/4]$  for an integer  $n$ , the windowed version  $y_{T,\tau}$  is in  $C^1$ . Figure 3.1(d) illustrates these conclusions with four windowed versions of  $x$ .



**Figure 3.2:** Functions illustrating the concept of bounded/unbounded variations. On the interval  $[0, 1]$ , only  $x_3(t)$  is of bounded variation.

**Space of Functions of Bounded Variation** Functions of bounded variation are easiest to understand when they are also continuous. A continuous function has bounded variation if the length of its graph on any finite interval is finite. While most of the functions we encounter satisfy this criterion, many do not. For example, consider the following functions:

$$x_1(t) = \sin(1/t), \quad x_2(t) = t \sin(1/t), \quad x_3(t) = t^2 \sin(1/t),$$

where each is defined to equal 0 for  $t = 0$ . On the interval  $[0, 1]$ ,  $x_3(t)$  is of bounded variation while  $x_1(t)$  and  $x_2(t)$  are not. Another example is a function  $x_4(t)$  defined on the unit interval  $[0, 1]$  and having value  $\pm 1$  over dyadic intervals:

$$x_4(t) = (-1)^i, \quad 2^{-i} \leq t < 2^{-i+1}, \quad i \in \mathbb{Z}^+, \quad t \in [0, 1],$$

or, equivalently,

$$x_4(t) = (-1)^{\lceil \log_2(1/t) \rceil}.$$

This function is not of bounded variation either. All four functions are shown in Figure 3.2.

Formally, the total variation of a function  $x$  over  $[a, b]$  is defined as:

$$V_a^b(x) = \sup_N \sup_{t_0, t_1, \dots, t_N} \sum_{k=0}^{N-1} |x(t_{k+1}) - x(t_k)|,$$

where the second supremum is taken over all increasing sequences  $(t_0, t_1, \dots, t_N)$  in  $[a, b]$ . Then, a real-valued  $x(t)$  is said to be of bounded variation over  $[a, b]$  when  $V_a^b(x)$  is finite.

### Special Functions

We now introduce the functions most often used in the book.

**Dirac Delta Function** The *Dirac delta function* satisfies

$$\delta(t) = 0 \quad \text{for all nonzero } t \quad (3.5a)$$

and

$$\int_{-\infty}^{\infty} \delta(t) dt = 1. \quad (3.5b)$$

Since no function can actually satisfy these properties, the Dirac delta is not a function and is sometimes called a *generalized function*. It is commonly used in signal processing to enable concise expressions that would otherwise require limits or integrations.

The Dirac delta function can be heuristically<sup>69</sup> related to a limit of a sequence of functions, where one suitable sequence is given by

$$d_n(t) = \begin{cases} n/2, & |t| \leq 1/n; \\ 0, & \text{otherwise,} \end{cases} \quad \text{for } n = 1, 2, \dots \quad (3.6)$$

While the sequence of functions does not converge in  $\mathcal{L}^2$  norm (see Exercise 3.1), we do have consistency with properties (3.5):

$$\lim_{n \rightarrow \infty} d_n(t) = 0 \quad \text{for any nonzero } t$$

and

$$\lim_{n \rightarrow \infty} \int_{-\infty}^{\infty} d_n(t) dt = 1.$$

Other properties of limits of integrals involving  $d_n$  are explored in Exercise 3.1, including an interpretation of derivatives of Dirac delta functions. Table 3.1 lists some properties of the Dirac delta function. (The shifting property uses convolution, which is defined in (3.36). For the sifting and sampling properties to hold,  $x(t)$  must be continuous at  $t_0$  and 0, respectively.)

Another function often used in signal processing, especially to describe sampling, is the *Dirac delta comb* or *picket fence* function. It is a sum of Dirac delta functions uniformly spaced at locations  $nT$ :

$$s_T(t) = \sum_{n \in \mathbb{Z}} \delta(t - nT). \quad (3.7)$$

<sup>69</sup>A formal derivation of the Dirac generalized function requires the knowledge of the theory of distributions. In particular, the Dirac function is defined indirectly through its action on a test function, such as the sifting property in Table 3.1. Thus, while (3.6) is an engineering trick, it is most useful in all nonpathological cases.

Dirac delta function	
Normalization	$\int_{-\infty}^{\infty} \delta(t) dt = 1$
Sifting	$\int_{-\infty}^{\infty} x(t_0 - t) \delta(t) dt = \int_{-\infty}^{\infty} x(t) \delta(t_0 - t) dt = x(t_0)$
Shifting	$x(t) * \delta(t - t_0) = x(t - t_0)$
Sampling	$x(t) \delta(t) = x(0) \delta(t)$
Restriction	$x(t) \delta(t) = 1_{\{0\}} x$

**Table 3.1:** Properties of the Dirac delta function.

**Heaviside Function** The *Heaviside* or *unit-step* function is defined as

$$u(t) = \begin{cases} 1, & t \geq 0; \\ 0, & \text{otherwise.} \end{cases} \quad t \in \mathbb{R}. \quad (3.8)$$

This function is bounded by 1, so it belongs to  $\mathcal{L}^\infty(\mathbb{R})$ . It belongs to neither  $\mathcal{L}^1(\mathbb{R})$  nor  $\mathcal{L}^2$ . The Dirac delta and Heaviside functions are related via

$$u(t) = \int_{-\infty}^t \delta(\tau) d\tau. \quad (3.9)$$

Pointwise multiplication by the Heaviside function implements the domain restriction operator (1.58) for restriction from all real numbers to just the nonnegative real numbers:

$$1_{\mathbb{R}^+} x = \begin{cases} x(t), & \text{for } t \geq 0; \\ 0, & \text{otherwise} \end{cases} = u(t) x(t), \quad t \in \mathbb{R}.$$

From this we can also build other domain restriction operators. For example, domain restriction to  $[t_0, t_1)$  is achieved with a difference of two shifted Heaviside functions:

$$1_{[t_0, t_1)} x = (u(t - t_0) - u(t - t_1)) x(t) = \begin{cases} x(t), & \text{for } t \in [t_0, t_1); \\ 0, & \text{otherwise.} \end{cases} \quad (3.10)$$

**Box Function** For any positive real number  $t_0$ , the *box* function is given by:<sup>70</sup>

$$\chi_{[-t_0/2, t_0/2]}(t) = \begin{cases} 1/\sqrt{t_0}, & |t| \leq t_0/2; \\ 0, & \text{otherwise,} \end{cases} \quad (3.11)$$

that is, it is an indicator function of the interval  $[-t_0/2, t_0/2]$ . It is of unit square norm and its integral  $\int_{t \in \mathbb{R}} \chi(t) dt = 1$  when  $t_0 = 1$ . The box function and the sinc function are intimately related; they form a Fourier-transform pair, as we will see later.

The box function can be expressed in terms of the Heaviside function as

$$\chi_{[-t_0/2, t_0/2]}(t) = u(t + \frac{1}{2}) - u(t - \frac{1}{2}). \quad (3.12)$$

<sup>70</sup>This is often called a *centered* and *normalized* box function.

**Gaussian Function** A Gaussian function is defined as

$$g(t) = \gamma e^{-\alpha(t-\mu)^2}, \quad (3.13a)$$

where  $\mu$  shifts the center of the function to  $t = \mu$ , and  $\alpha$  and  $\gamma$  are positive constants.

When  $\alpha = 1/(2\sigma^2)$  and  $\gamma = 1/(\sigma\sqrt{2\pi})$ ,  $\|g\|_1 = 1$ , and thus  $g$  can be seen as a probability density function, with  $\mu$  and  $\sigma$  interpreted as the mean and standard deviation, respectively:

$$\begin{aligned} \|g(t)\|_1 &= \int_{-\infty}^{\infty} |g(t)| dt = \int_{-\infty}^{\infty} \gamma e^{-\alpha(t-\mu)^2} dt \\ &= \int_{-\infty}^{\infty} \frac{1}{\sigma\sqrt{2\pi}} e^{-(t-\mu)^2/(2\sigma^2)} dt = 1. \end{aligned} \quad (3.13b)$$

When  $\gamma = (2\alpha/\pi)^{1/4}$ ,  $\|g\| = 1$ , that is,  $g$  is of unit norm (energy):

$$\begin{aligned} \|g(t)\|_2^2 &= \int_{-\infty}^{\infty} |g(t)|^2 dt = \int_{-\infty}^{\infty} \gamma^2 e^{-2\alpha(t-\mu)^2} dt \\ &= \int_{-\infty}^{\infty} \sqrt{\frac{2\alpha}{\pi}} e^{-2\alpha(t-\mu)^2} dt = 1. \end{aligned} \quad (3.13c)$$

### Deterministic Correlation

We now discuss two operations on functions, both deterministic, that appear throughout the chapter; these are analogous to the notions of deterministic autocorrelation and crosscorrelation for sequences defined in Section 2.2. Stochastic versions of both operations will be given in Section 3.6.1.

**Deterministic Autocorrelation** The *deterministic autocorrelation*  $a$  of a function  $x$  is

$$a(t) = \int_{-\infty}^{\infty} x(\tau) x^*(\tau - t) d\tau = \langle x(\tau), x(\tau - t) \rangle_{\tau}. \quad (3.14)$$

The deterministic autocorrelation satisfies

$$a(t) = a^*(-t), \quad (3.15a)$$

$$a(0) = \int_{-\infty}^{\infty} |x(\tau)|^2 d\tau = \|x\|^2, \quad (3.15b)$$

analogously to (2.17). The deterministic autocorrelation measures the similarity of a function with respect to shifts of itself, and it is Hermitian symmetric as in (3.15a). For a real  $x$ ,

$$a(t) = \int_{-\infty}^{\infty} x(\tau) x(\tau - t) d\tau = a(-t). \quad (3.15c)$$

When we need to specify the function involved, we write  $a_x(t)$ .

**Deterministic Crosscorrelation** The *deterministic crosscorrelation*  $c$  of two functions  $x$  and  $y$  is

$$c(t) = \int_{-\infty}^{\infty} x(\tau) y^*(\tau - t) d\tau = \langle x(\tau), y(\tau - t) \rangle_{\tau}, \quad (3.16)$$

and is written as  $c_{x,y}(t)$  to specify the functions involved. It satisfies

$$c_{x,y}(t) = \left( \int_{-\infty}^{\infty} y(\tau - t) x^*(\tau) d\tau \right)^* \stackrel{(a)}{=} \left( \int_{-\infty}^{\infty} y(\tau') x^*(\tau' + t) d\tau' \right)^* = c_{y,x}^*(-t), \quad (3.17a)$$

where (a) follows from change of variable  $\tau' = \tau - t$ . For real  $x$  and  $y$ ,

$$c_{x,y}(t) = \int_{-\infty}^{\infty} x(\tau) y(\tau - t) d\tau = c_{y,x}(-t). \quad (3.17b)$$

### 3.2.2 Periodic Functions

Periodic functions with period  $T$  satisfy

$$x(t + T) = x(t), \quad t \in \mathbb{R}. \quad (3.18)$$

Such functions appear in many physical problems, most notably in the original work of Fourier on heat conduction in a circular wire. In general, such functions do not have a finite  $\mathcal{L}^1$  or  $\mathcal{L}^2$  norm; instead, we consider functions such that a period is in  $\mathcal{L}^1$  or  $\mathcal{L}^2$ , respectively:

$$\int_{-T/2}^{T/2} |x(t)| dt < \infty \quad \text{or} \quad \int_{-T/2}^{T/2} |x(t)|^2 dt < \infty. \quad (3.19)$$

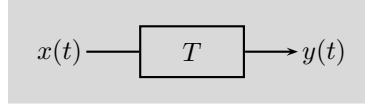
As we said earlier, another way to look at these functions is as functions on an interval, circularly extended, similarly to what we have seen in Chapter 2.

### 3.2.3 Multidimensional Functions

Multidimensional functions are functions in several variables. A two-dimensional example is a function  $x(t_1, t_2)$ . While they can have an arbitrary number of dimensions, in signal processing, two- and three-dimensional functions are typical, such as images (two dimensions), or video (three dimensions). An obvious generalization of the one-dimensional case is when these functions are separable. For example, in two dimensions, separable functions have the form

$$x(t_1, t_2) = x_1(t_1)x_2(t_2). \quad (3.20)$$

Clearly, one can apply one-dimensional theory to each factor, such as the factorization of polynomials. While limited, this separable class is quite popular due to its simplicity. For nonseparable functions of multiple variables, certain techniques used in one dimension do not generalize to multiple dimensions. For example, the fundamental theorem of algebra, Theorem 2.19, is stated for polynomials of one variable only. The notion of norms extends directly to multidimensional functions, as do the smoothness classes.

**Figure 3.3:** A continuous-time system.

### 3.3 Systems

Continuous-time systems are operators having continuous-time functions as their inputs and outputs. Among all continuous-time systems, we will concentrate on those that are linear and shift-invariant. This subclass is both important in practice and amenable to easy analysis. After an introduction to differential equations, which are natural descriptions of continuous-time systems, we study linear, shift-invariant systems in detail.

#### 3.3.1 Continuous-Time Systems and Their Properties

A continuous-time system is an operator  $T$  that maps an input function  $x \in V$  into an output function  $y \in V$ ,

$$y = T(x), \quad (3.21)$$

as shown in Figure 3.3. As we have seen in the previous section, the function space  $V$  is typically  $\mathcal{L}^2(\mathbb{R})$  or  $\mathcal{L}^\infty(\mathbb{R})$ . At times, the input or the output is in a subspace of such spaces.

#### Types of Systems

For each of the types of systems seen in the previous chapter, we have a continuous-time counterpart. As the concepts are identical, we just list them for completeness. It is instructive to compare the discrete-time ones against the continuous-time ones.

**Linear Systems** The definition of linearity of a continuous-time system is similar to Definition 1.17 of a linear operator and Definition 2.1 of a linear discrete-time system:

**DEFINITION 3.1 (LINEAR SYSTEM)** A continuous-time system  $T$  is called linear when, for any inputs  $x$  and  $y$  and any  $\alpha, \beta \in \mathbb{C}$ ,

$$T(\alpha x + \beta y) = \alpha T(x) + \beta T(y). \quad (3.22)$$

The function  $T$  is thus a linear operator, and we write (3.21) as

$$y = T x. \quad (3.23)$$

**Memoryless Systems** The definition of a memoryless system closely follows Definition 2.2.

DEFINITION 3.2 (MEMORYLESS SYSTEM) A continuous-time system  $T$  is called memoryless when, for any real  $\tau$  and inputs  $x$  and  $x'$ ,

$$1_{\{\tau\}} x = 1_{\{\tau\}} x' \Rightarrow 1_{\{\tau\}} T(x) = 1_{\{\tau\}} T(x'). \quad (3.24)$$

**Causal Systems** The output of a causal system at time  $t$  depends on the input only up to time  $t$ . If two inputs agree up to time  $k$ , the corresponding outputs must agree up to time  $k$ :

DEFINITION 3.3 (CAUSAL SYSTEM) A continuous-time system  $T$  is called causal when, for any real  $\tau$  and inputs  $x$  and  $x'$ ,

$$1_{(-\infty, \tau]} x = 1_{(-\infty, \tau]} x' \Rightarrow 1_{(-\infty, \tau]} T(x) = 1_{(-\infty, \tau]} T(x'). \quad (3.25)$$

As discussed in Section 2.3.1, causality can seem to be a property that is required of any real system. When the time variable literally represents time and the same time origin is used for the input and output functions, causality is indeed necessary for accurate models of physical systems.

**Shift-Invariant Systems** In a shift-invariant system, shifting the input has the effect of shifting the output by the same amount:

DEFINITION 3.4 (SHIFT-INVARIANT SYSTEM) A continuous-time system  $T$  is called shift invariant when, for any real  $\tau$  and input  $x$ ,

$$y = T(x) \Rightarrow y' = T(x'), \quad \text{where } x'(t) = x(t - \tau) \text{ and } y'(t) = y(t - \tau). \quad (3.26)$$

**Stable Systems** As for discrete-time systems, we define and consider *bounded input bounded output (BIBO)* stability exclusively:

DEFINITION 3.5 (BIBO STABILITY) A continuous-time system  $T$  is called bounded-input bounded-output stable when a bounded input  $x$  produces a bounded output  $y = T(x)$ :

$$x \in \mathcal{L}^\infty(\mathbb{R}) \Rightarrow y \in \mathcal{L}^\infty(\mathbb{R}). \quad (3.27)$$



### Basic Systems

We now discuss a few basic continuous-time systems.

**Shift** The shift-by- $t_0$  operator is defined as:

$$y(t) = x(t - t_0), \quad t \in \mathbb{R}, \quad (3.28)$$

and simply delays  $x(t)$  by  $t_0$ . It is an LSI operator, causal and BIBO stable, but not memoryless. While this is one of the simplest continuous-time systems, it is also the most important, as the whole concept of time processing is based on this simple operator. Compare this continuous-time shift operator to the discrete-time one defined in (2.38).

**Modulator** While the shift we just saw is the shift in time, modulation is shift in frequency (as we will see later in this chapter). A modulation by a complex exponential of frequency  $\omega_0$  is given by

$$y(t) = e^{j\omega_0 t} x(t), \quad t \in \mathbb{R}. \quad (3.29)$$

This operator is linear, causal, memoryless and BIBO stable, but not shift invariant. For those already familiar with Fourier analysis, (3.29) shifts the spectrum of  $x(t)$  to the position  $\omega_0$  in frequency. Compare this continuous-time modulation operator to the discrete-time one defined in (2.40).

**Integrator** Similarly to the accumulator (2.42) in discrete time, an integrator sums up the inputs up to the present time:

$$y(t) = \int_{-\infty}^t x(\tau) d\tau, \quad t \in \mathbb{R}. \quad (3.30)$$

This is an LSI, causal operator, but not memoryless nor BIBO stable.

**Averaging Operators** As in (2.46), for any fixed  $T > 0$ , we could consider a system that takes an average of the input,

$$y(t) = \frac{1}{T} \int_{t-T/2}^{t+T/2} x(\tau) d\tau, \quad t \in \mathbb{R}. \quad (3.31)$$

This is a *moving average* filter since we look at the function through a window of length  $T$ . This operator is LSI and BIBO stable, but neither memoryless nor causal.

We could obtain a causal version by simply delaying the moving average in (3.31) by  $T/2$ , resulting in

$$y(t) = \frac{1}{T} \int_{t-T}^t x(\tau) d\tau, \quad t \in \mathbb{R}. \quad (3.32)$$

This operator is again LSI and BIBO stable but also causal, while still not memoryless.

**Maximum Operator** This simple operator computes the maximum value of the input up to the current time:

$$y(t) = \max(1_{\{-\infty, \dots, t\}} x). \quad (3.33)$$

This operator is clearly neither linear nor memoryless, but it is causal, shift invariant, and BIBO stable.

### 3.3.2 Differential Equations

While differential equations are typically encountered before difference equations, in our exposition this is not the case. In Chapter 2, we have examined the basic principles behind difference equations. As in discrete time, where linear difference equations relate the input sequence and past outputs to the current output, in continuous time, linear differential equations relate the input function and past outputs to the current output. In particular, linear constant-coefficient differential equations (compare to linear constant-coefficient difference equations in (2.54)) describe LSI systems and are of the form:

$$y(t) = \sum_{k=0}^M b_k \frac{d^k x(t)}{dt^k} - \sum_{k=1}^N a_k \frac{d^k y(t)}{dt^k}. \quad (3.34)$$

To find the solution, we follow the procedure outlined in Section 2.A.2. (1) We find a solution,  $y^{(h)}(t)$ , to the homogeneous equation by setting the input  $x(t)$  in (3.34) to zero. (2) We then find any particular solution,  $y^{(p)}(t)$ , to (3.34), typically by assuming the output of the same form as the input. (3) Finally, the complete solution is formed by superposition of the solution to the homogeneous equation and the particular solution. The coefficients in the homogeneous solution are found by specifying initial conditions for  $y(t)$  and then solving the system. A standard way of finding solutions to differential equations is to use Fourier and Laplace transforms, and thus, we will revisit differential equations once we are equipped with these tools.

### 3.3.3 Linear Shift-Invariant Systems

#### Impulse Response

The impulse response of an LSI continuous-time system is defined with the Dirac delta function playing the role that the Kronecker delta sequence plays in discrete time:

**DEFINITION 3.6 (IMPULSE RESPONSE)** A function  $h$  is called the impulse response of LSI continuous-time system  $T$  when input  $\delta$  produces output  $h$ .

The impulse response  $h$  of a causal linear system always satisfies  $h(t) = 0$  for all  $t < 0$ . This is required because, according to (3.25), the output in response to input

## 3.3. Systems

331

$\delta$  must match on  $(-\infty, 0)$  to the  $\mathbf{0}$  output function that results from the  $\mathbf{0}$  input function.

An LSI system is completely specified by its impulse response. In discrete time, this result is simple because the Kronecker delta sequence and its shifts forms a basis; see (2.58). We shall not formally prove the continuous-time result here, but rather provide an intuitive explanation.

Assume we can approximate the function  $x(t)$  by a linear combination of  $d_n(t)$  from (3.6) and its shifts by integer multiples of  $2/n$ :

$$x_n(t) \approx \sum_{k \in \mathbb{Z}} \frac{2}{n} x(2k/n) d_n(t - 2k/n). \quad (3.35a)$$

Denote the response of the system to input  $d_n(t)$  by  $h_n(t)$ . Then similarly to (2.58), the response of the system to  $x_n(t)$  is expressed by linear combination of  $h_n(t)$  and its shifts by integer multiples of  $2/n$ :

$$y_n(t) = \sum_{k \in \mathbb{Z}} \frac{2}{n} x(2k/n) h_n(t - 2k/n). \quad (3.35b)$$

Now taking  $n \rightarrow \infty$  causes  $d_n \rightarrow \delta$  and  $h_n \rightarrow h$  (by definition). The sums in (3.35) become Riemann integrals, so  $x_n \rightarrow x$  by the sifting property in Table 3.1, and  $y_n \rightarrow y$  where  $y$  is determined by the input and the impulse response  $h$ , as we wished to demonstrate. While the use of the Dirac delta function and presumptions of convergence make this merely a heuristic argument, it does capture the main ideas.

**Convolution**

To parallel (2.58), we can express an arbitrary input to LSI system  $T$  as

$$x(t) = \int_{-\infty}^{\infty} x(\tau) \delta(t - \tau) d\tau,$$

by using the sifting property in Table 3.1. Then the output resulting from this input is

$$\begin{aligned} y &= Tx = T \int_{-\infty}^{\infty} x(\tau) \delta(t - \tau) d\tau \stackrel{(a)}{=} \int_{-\infty}^{\infty} x(\tau) T\delta(t - \tau) d\tau \\ &\stackrel{(b)}{=} \int_{-\infty}^{\infty} x(\tau) h(t - \tau) d\tau = h * x, \end{aligned}$$

where (a) follows from linearity; and (b) from shift invariance and the definition of impulse response, defining the convolution:

**DEFINITION 3.7 (CONVOLUTION)** The convolution between functions  $h$  and  $x$  is defined as

$$(Hx)(t) = (h * x)(t) = \int_{-\infty}^{\infty} x(\tau) h(t - \tau) d\tau, \quad (3.36)$$

where  $H$  is called the convolution operator associated with  $h$ .

The convolution equation reduces the problem of solving ordinary differential equations to that of finding impulse responses.

**Properties** The convolution (3.36) satisfies:

(i) *Connection to the inner product*

$$(h * x)(t) = \int_{-\infty}^{\infty} x(\tau) h(t - \tau) d\tau = \langle x(\tau), h^*(t - \tau) \rangle_{\tau}. \quad (3.37a)$$

(ii) *Commutativity*

$$h * x = x * h. \quad (3.37b)$$

(iii) *Associativity*

$$g * (h * x) = g * h * x = (g * h) * x. \quad (3.37c)$$

(iv) *Deterministic autocorrelation*

$$a(t) = \int_{-\infty}^{\infty} x(\tau) x^*(\tau - t) d\tau = x(t) *_t x^*(-t). \quad (3.37d)$$

These properties have discrete-time counterparts in (2.61) and are explored further in Solved Exercise 3.2. Note that all the properties of convolution depend on the integrals—whether written explicitly or implicitly—converging. We will not dwell on these technicalities; Appendix 2.A.3, though focused only on discrete time, has a related discussion.

**Filters** As in Chapter 2, the impulse response is often called a *filter* and the convolution is called *filtering*.

**Stability** Similarly to Chapter 2, BIBO stability for continuous-time systems is equivalent to the absolute integrability of the impulse response. The proof is similar to the discrete-time case (see Proposition 2.8) and is left for Exercise 3.2.

**PROPOSITION 3.8 (BIBO STABILITY)** An LSI system is BIBO stable if and only if its impulse response  $h(t)$  is absolutely integrable.

**Smoothing** One key feature of many convolution operators is their smoothing effect. For example, when the impulse response has a nonzero mean (zeroth moment, see (3.63a)), the convolution will compute a local average, as we now show:

**EXAMPLE 3.2 (LOCAL SMOOTHING BY CONVOLUTION)** Choose the box function of width  $t_0 > 0$ , (3.11), as the impulse response  $h(t)$ , and a piecewise-constant function over integer intervals,

$$x(t) = x_n, \quad \text{for } t \in [n, n+1), \quad n \in \mathbb{Z},$$

as the input (for some sequence  $x_n$ ). The convolution  $y = h * x$ , continuous for any  $t_0$ . For example, the output with  $t_0 = 1$  is

$$y(t) = x_n + (t - n - \frac{1}{2})(x_{n+1} - x_n), \quad \text{for } t \in [n + \frac{1}{2}, n + \frac{3}{2}], \quad n \in \mathbb{Z},$$

which is piecewise linear and continuous. Thus, thanks to a smoothing impulse response, a discontinuous function  $x$  is transformed into a  $C^0$  function  $y$ .

### Circular Convolution

We now consider what happens with our second class of functions, those that are either periodic or of finite length circularly extended. To start, we assume that the impulse response  $h$  is in  $\mathcal{L}^1(\mathbb{R})$ .

**Linear Convolution with Circularly-Extended Signal** Given a bounded periodic function  $x(t)$  as in (3.18) and a filter with impulse response  $h(t)$  in  $\mathcal{L}^1(\mathbb{R})$ , we can compute the convolution as usual:

$$y(t) = (h * x)(t) = \int_{-\infty}^{\infty} x(\tau)h(t - \tau) d\tau = \int_{-\infty}^{\infty} h(\tau)x(t - \tau) d\tau. \quad (3.39)$$

Since  $x(t)$  is  $T$ -periodic,  $y(t)$  is  $T$ -periodic as well:

$$y(t + T) = \int_{-\infty}^{\infty} h(\tau)x(t + T - \tau) d\tau \stackrel{(a)}{=} \int_{-\infty}^{\infty} h(\tau)x(t - \tau) d\tau = y(t),$$

where (a) follows from the periodicity of  $x$ .

Let us now define a periodized version of  $h(t)$  as:

$$h_T(t) = \sum_{n \in \mathbb{Z}} h(t - nT), \quad (3.40)$$

which converges for every  $t$  because  $h \in \mathcal{L}^1(\mathbb{R})$ . The new function  $h_T$  is  $T$ -periodic with a period in  $\mathcal{L}^1(\mathbb{R})$ . We now want to show how we can express the convolution

in (3.39) in terms of what we will define as a *circular* convolution:

$$\begin{aligned}
 (h * x)(t) &= \int_{-\infty}^{\infty} h(\tau)x(t - \tau) d\tau \stackrel{(a)}{=} \sum_{n \in \mathbb{Z}} \int_{nT-T/2}^{nT+T/2} h(\tau)x(t - \tau) d\tau \\
 &\stackrel{(b)}{=} \sum_{n \in \mathbb{Z}} \int_{-T/2}^{T/2} h(\tau' + nT)x(t - \tau' - nT) d\tau' \\
 &\stackrel{(c)}{=} \sum_{n \in \mathbb{Z}} \int_{-T/2}^{T/2} h(\tau + nT)x(t - \tau) d\tau \\
 &\stackrel{(d)}{=} \int_{-T/2}^{T/2} \underbrace{\sum_{n \in \mathbb{Z}} h(\tau + nT)}_{h_T(\tau)} x(t - \tau) d\tau \\
 &= \int_{-T/2}^{T/2} h_T(\tau)x(t - \tau) d\tau = (h_T \circledast x)(t), \tag{3.41}
 \end{aligned}$$

where in (a) we split the real line into segments of length  $T$ ; (b) follows from change of variable  $\tau' = \tau - nT$ ; (c) follows from periodicity of  $x$  and change of variable  $\tau = \tau'$ ; and (d) follows from (1.196). The expression above tends to be more convenient as it involves only one period of both  $x$  and the periodized version  $h_T$  of the impulse response  $h$ .

**Definition of the Circular Convolution** In computing the convolution of a periodic function  $x$  with an impulse response  $h \in \mathcal{L}^1(\mathbb{R})$ , we implicitly defined the *circular convolution* of a  $T$ -periodic function  $x$  and a  $T$ -periodic impulse response  $h$ :

DEFINITION 3.9 (CIRCULAR CONVOLUTION) The circular convolution between  $T$ -periodic functions  $h$  and  $x$  is defined as

$$(Hx)(t) = (h \circledast x)(t) = \int_{-T/2}^{T/2} h(\tau)x(t - \tau) d\tau, \tag{3.42}$$

where  $H$  is called the circular convolution operator associated with  $h$ .

The result of the circular convolution is again  $T$ -periodic. While this notion of convolution is independent from that of linear convolution, we have just seen that the two are related when the input function is periodic but the impulse response of the system is not. We made the connection by periodizing that impulse response.

### 3.4 Fourier Transform

As discussed in Section 2.4 as well as at the beginning of this chapter, the ubiquity of the Fourier transform is mostly due to the fact that the complex exponentials are

eigenfunctions of LSI systems (convolution operators), and operating in the spaces defined by those eigenfunctions is a natural representation for LSI systems. These facts, as in discrete time, lead to the convolution property, which states that the convolution operator is diagonalized by the Fourier transform. In this section, we review the Fourier transform for functions on the real line.

### 3.4.1 Definition of the Fourier Transform

**Eigenfunctions of the Convolution Operator** We now expand on what we have seen in the introduction as well as in Chapter 2; we demonstrate a fundamental property of LSI systems: complex exponential functions  $v_\lambda$  are eigenfunctions of the convolution operator  $H$ ,

$$H v_\lambda = h * v_\lambda = \lambda v_\lambda, \quad (3.43)$$

with the convolution operator as in (3.36). As before, to prove this, we must find these  $v_\lambda$  and the corresponding eigenvalues  $\lambda$ . Not surprisingly, the complex exponential function

$$v_\omega(t) = e^{j\omega t} \quad (3.44)$$

generates an entire space  $S_\omega = \{\alpha e^{j\omega t} \mid \alpha \in \mathbb{C}, \omega \in \mathbb{R}\}$ . The quantity  $\omega$  is called *angular frequency*; it is measured in radians per second. With  $\omega = 2\pi f$ , the quantity  $f$  is called *frequency*; it is measured in Hertz, or the number of cycles per second.

Let us now check what happens if we convolve  $h$  with  $v_\omega$  as in (3.43):

$$\begin{aligned} H v_\omega &= h * v_\omega = \int_{-\infty}^{\infty} v_\omega(t - \tau) h(\tau) d\tau \\ &= \int_{-\infty}^{\infty} e^{j\omega(t - \tau)} h(\tau) d\tau = \underbrace{\int_{-\infty}^{\infty} h(\tau) e^{-j\omega\tau} d\tau}_{\lambda_\omega} \underbrace{e^{j\omega t}}_{v_\omega}. \end{aligned} \quad (3.45)$$

Indeed, applying the convolution operator  $H$  to the complex exponential function  $v_\omega(t) = e^{j\omega t}$  results in the same function, albeit scaled by the corresponding eigenvalue  $\lambda_\omega$ . We call that eigenvalue the *frequency response* of the system. Although we have already seen it in the introduction, we repeat it here for completeness:

$$H(\omega) = \lambda_\omega = \int_{-\infty}^{\infty} h(\tau) e^{-j\omega\tau} d\tau. \quad (3.46)$$

We can thus rewrite (3.45) as

$$H e^{j\omega t} = h * e^{j\omega t} = H(\omega) e^{j\omega t}. \quad (3.47)$$

The above is true for any  $v_\omega \in S_\omega$ , and thus, that space does not change (it is invariant) under the operation of convolution.

**Fourier Transform** We are now ready to define the Fourier transform, which amounts to projecting  $x$  onto each each of the  $S_\omega$ :

**DEFINITION 3.10 (FOURIER TRANSFORM)** The Fourier transform of a function  $x(t)$  is

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt, \quad \omega \in \mathbb{R}. \quad (3.48a)$$

It exists when (3.48a) converges for all  $\omega \in \mathbb{R}$ ; we then call it the *spectrum* of  $x$ . The inverse Fourier transform of  $X(\omega)$  is

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega)e^{j\omega t} d\omega, \quad t \in \mathbb{R}. \quad (3.48b)$$

When the Fourier transform exists, we denote the Fourier-transform pair as

$$x(t) \xleftrightarrow{\text{FT}} X(\omega).$$

### 3.4.2 Existence and Convergence of the Fourier Transform

The existence, convergence, and inversion of the Fourier transform of a function will depend strongly on its properties, such as to which space it belongs. We will concentrate on basic cases where precise statements can be made without too much technical baggage. However, it should be understood that the validity of many of the relations we state is much wider; this will be indicated later.

**Functions in  $\mathcal{L}^1(\mathbb{R})$**  If  $x \in \mathcal{L}^1(\mathbb{R})$ , then  $X(\omega)$  converges pointwise, and  $X(\omega)$  is bounded and continuous (Solved Exercise 3.3), and the inversion formula (3.48b) holds. Since  $X(\omega) \in \mathcal{L}^1(\mathbb{R})$ ,  $x(t)$  is also bounded and continuous. The proof of the inversion formula for  $x \in \mathcal{L}^1(\mathbb{R})$  is somewhat technical and can be found in [100].

**EXAMPLE 3.3 (FOURIER TRANSFORM OF THE HAT FUNCTION)** Consider the hat function:

$$x(t) = \begin{cases} 1 - |t|, & |t| < 1; \\ 0, & \text{otherwise.} \end{cases} \quad (3.49a)$$

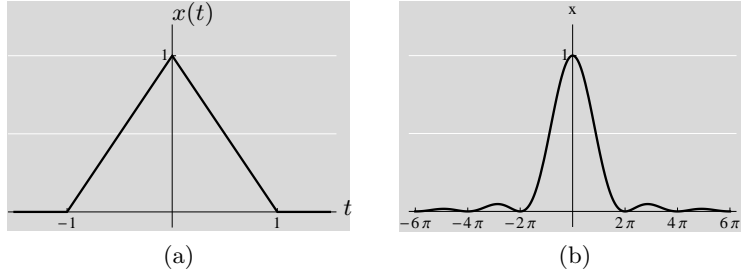
Its Fourier transform is

$$X(\omega) = \int_{-1}^0 (t+1)e^{-j\omega t} dt + \int_0^1 (1-t)e^{-j\omega t} dt. \quad (3.49b)$$

Pulling the constant terms together:

$$\int_{-1}^1 e^{-j\omega t} dt = -\frac{1}{j\omega} e^{-j\omega t} \Big|_{-1}^1 = \frac{e^{j\omega} - e^{-j\omega}}{j\omega}. \quad (3.49c)$$





**Figure 3.4:** The hat function (a) and its Fourier transform (b).

Then,

$$\begin{aligned} \int_0^1 (-t)e^{-j\omega t} dt &\stackrel{(a)}{=} \frac{1}{\omega^2} \int_0^{-j\omega} ue^u du \stackrel{(b)}{=} \frac{1}{\omega^2} (u-1)e^u \Big|_0^{-j\omega} \\ &= \frac{e^{-j\omega}}{j\omega} - \frac{e^{-j\omega}}{\omega^2} + \frac{1}{\omega^2}, \end{aligned} \quad (3.49d)$$

where (a) follows from change of variable  $u = -j\omega t$ ; and (b) from the fact that the primitive of  $ue^u$  is  $(u-1)e^u$ .<sup>71</sup> By similar arguments,

$$\int_{-1}^0 te^{-j\omega t} dt = -\frac{e^{j\omega}}{j\omega} - \frac{e^{j\omega}}{\omega^2} + \frac{1}{\omega^2}. \quad (3.49e)$$

Summing (3.49c)–(3.49e), we find  $X(\omega)$  from (3.49b) as

$$X(\omega) = \frac{1}{\omega^2} (2 - e^{j\omega} - e^{-j\omega}) = \left( \frac{e^{j\omega/2} - e^{-j\omega/2}}{j\omega} \right)^2 = \left( \text{sinc} \left( \frac{\omega}{2} \right) \right)^2. \quad (3.49f)$$

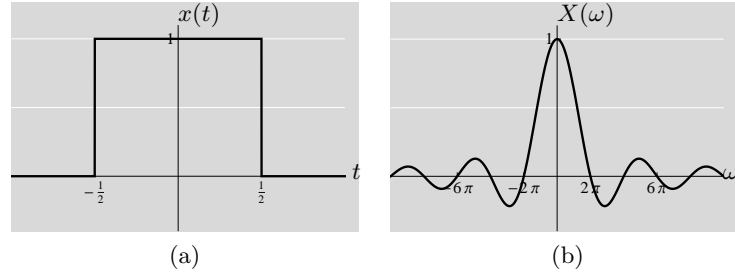
This Fourier transform is absolutely integrable and is thus in  $\mathcal{L}^1(\mathbb{R})$ . The hat function and its Fourier transform are depicted in Figure 3.4.

The hat function can also be seen as the convolution of two box functions from (3.11) with  $t_0 = 1$ ; as we will see in Example 3.4, the Fourier transform of such a box function is  $\text{sinc}(\omega/2)$ . Then, by convolution property (3.64), the Fourier transform of a convolution is the product of the Fourier transforms, leading to (3.49f).

**Functions in  $\mathcal{L}^2(\mathbb{R})$**  If  $x \in \mathcal{L}^2(\mathbb{R})$ , the inversion formula (3.48b) holds. Moreover, the  $\mathcal{L}^2$  norm is conserved (up to a constant), since

$$\|x(t)\|^2 = \frac{1}{2\pi} \|X(\omega)\|^2. \quad (3.50)$$

<sup>71</sup>A function  $x(t)$  is a primitive of  $y(t)$ , if  $x'(t) = y(t)$ . We will denote that primitive function with a superscript <sup>(1)</sup>, as in  $y^{(1)}(t)$ .



**Figure 3.5:** The (a) box function and (b) its Fourier transform (for  $t_0 = 1$ ).

This is the Parseval's equality<sup>72</sup> from Chapter 1, and is formally given in (3.69a).

The extension from  $\mathcal{L}^1$  to  $\mathcal{L}^2$  is technically nontrivial, since when  $x(t)$  is not absolutely integrable, then neither is  $x(t)e^{-j\omega t}$ . We refer to [100, 19] for a thorough discussion of this topic, and only prove Parseval's equality for functions that are in both  $\mathcal{L}^1$  and  $\mathcal{L}^2$ , later in this chapter.

**EXAMPLE 3.4 (FOURIER TRANSFORM OF THE BOX FUNCTION)** We now derive the Fourier transform of the box function from (3.11):

$$\begin{aligned} X(\omega) &= \frac{1}{\sqrt{t_0}} \int_{-t_0/2}^{t_0/2} e^{-j\omega t} dt = -\frac{1}{j\omega\sqrt{t_0}} e^{-j\omega t} \Big|_{-t_0/2}^{t_0/2} \\ &= \frac{e^{j\omega t_0/2} - e^{-j\omega t_0/2}}{j\omega\sqrt{t_0}} = \sqrt{t_0} \operatorname{sinc}\left(\frac{\omega t_0}{2}\right). \end{aligned} \quad (3.51a)$$

This function is not absolutely integrable, since it decays only as  $O(1/|1 + \omega|)$ . By Parseval's equality, it is in  $\mathcal{L}^2(\mathbb{R})$ , however, allowing us to use  $\mathcal{L}^2$  inversion.<sup>73</sup> The function and its Fourier transform for  $t_0 = 1$  are shown in Figure 3.5. Using (2.8c), we see that the Fourier transform of the box function is zero at all integer multiples of  $\omega = 2\pi/t_0$ :

$$X\left(\frac{2k\pi}{t_0}\right) = \sqrt{t_0}\delta_k. \quad (3.51b)$$

As we already mentioned in Example 3.3, the hat function (3.49a) is simply the convolution of the box function (when  $t_0 = 1$ ) with itself; thus,

$$x * x \xleftrightarrow{\text{FT}} X^2(\omega) = \left(\operatorname{sinc}\left(\frac{\omega}{2}\right)\right)^2,$$

that is, its Fourier transform can be obtained from the Fourier transform of the box function using the convolution property (3.64).

<sup>72</sup>Recall that what we call Parseval's equality in this book is sometimes called Plancherel's equality as well; what we call generalized Parseval's equality is Parseval's theorem.

<sup>73</sup>At points of discontinuity, the inversion leads to a midpoint reconstruction.

**EXAMPLE 3.5 (FOURIER TRANSFORM OF THE SINC FUNCTION)** We just looked at the box function in time; the dual case is the box function in frequency, which is used to represent an ideal lowpass filter. Such a filter keeps frequencies between  $-\omega_0/2$  and  $\omega_0/2$  perfectly while suppressing all others, or

$$H(\omega) = \begin{cases} \sqrt{2\pi/\omega_0}, & |\omega| \leq \omega_0/2; \\ 0, & \text{otherwise.} \end{cases} \quad (3.52a)$$

Except for scaling, this is dual to Example 3.4, and thus

$$h(t) = \frac{1}{\sqrt{2\pi\omega_0}} \int_{-\omega_0/2}^{\omega_0/2} e^{j\omega t} d\omega = \sqrt{\frac{\omega_0}{2\pi}} \operatorname{sinc}\left(\frac{\omega_0 t}{2}\right). \quad (3.52b)$$

As in its discrete-time counterpart in Table 2.5, the factor  $\sqrt{\omega_0/2\pi}$  is present to make  $h(t)$  unit norm. Using (2.8c):

$$h\left(\frac{2k\pi}{\omega_0}\right) = \sqrt{\frac{\omega_0}{2\pi}} \delta_k, \quad (3.52c)$$

that is,  $h(t)$  is zero at all integer multiples of  $T = 2\pi/\omega_0$ , except at the origin.

**Convergence** The history of Fourier analysis and Fourier series is marked by results showing potential problems with convergence. While for continuous functions in  $\mathcal{L}^1$  or  $\mathcal{L}^2$  things are relatively straightforward, not so for functions from other spaces. Yet, the construction of such functions (for example, Weierstrass's continuous, but nowhere differentiable function), led to a more fundamental understanding of the notion of a function, but also of subtle concepts such as Brownian motion, studied at the time.

Various forms of convergence of the Fourier transform or its inverse are possible; the three main ones are pointwise convergence, uniform convergence and convergence in norm (see Appendix 1.A.2). A question of both theoretical and practical interest is the following: If a function  $x(t)$  has Fourier transform  $X(\omega)$ , and we compute the inverse Fourier transform  $\hat{x}(t)$  from  $X(\omega)$ , when is  $\hat{x}(t) = x(t)$ , and in what sense (for example, almost everywhere, in norm)?

- (i) If  $x(t) \in \mathcal{L}^1(\mathbb{R})$  and  $X(\omega) \in \mathcal{L}^1(\mathbb{R})$ , then  $\hat{x}(t) = x(t)$  almost everywhere. If in addition,  $x(t)$  is continuous, then

$$\hat{x}(t) = x(t) \quad \text{for all } t. \quad (3.53a)$$

This follows from  $\mathcal{L}^1$  inversion, by adding the fact that two continuous functions that are almost everywhere equal are necessarily equal everywhere.

- (ii) If  $x(t) \in \mathcal{L}^2(\mathbb{R})$ , then

$$\|x(t) - \hat{x}(t)\| = 0, \quad (3.53b)$$

since the Fourier transform is a unitary map (see (3.50)).

A single discontinuity in a function leads to a Fourier transform that is nonintegrable. Its inverse Fourier transform does not converge uniformly, and this behavior that is both famous and annoying, was first described by Gibbs in the late 19th century. We have already seen an example in Figure 0.3; for the discrete version of this phenomenon, see Section 2.4.2.

**EXAMPLE 3.6 (GIBBS PHENOMENON)** Consider a function  $x(t)$  smooth everywhere except for a single, step discontinuity. Without loss of generality, assume this discontinuity to be at the origin and of height 1. Therefore, we can write  $x(t)$  as

$$x(t) = x_s(t) + u(t) \quad (3.54a)$$

where  $x_s(t)$  is smooth everywhere and  $u(t)$  is the Heaviside function (3.8) (see Figure 3.6).

Now consider  $X(\omega)$  and its restriction to  $[-\omega_0/2, \omega_0/2]$ ,

$$X_{\omega_0}(\omega) = X|_{[-\omega_0/2, \omega_0/2]}(\omega) = \begin{cases} X(\omega), & |\omega| < \omega_0/2; \\ 0, & \text{otherwise.} \end{cases} \quad (3.54b)$$

This can be seen as a lowpass version of  $X(\omega)$ , that is,

$$x_{\omega_0} = h_{\omega_0} * x = h_{\omega_0} * (x_s + u) = h_{\omega_0} * x_s h_{\omega_0} + \underbrace{h_{\omega_0} * u}_{u_{\omega_0}}, \quad (3.54c)$$

where  $h_{\omega_0}(t)$  is as in (3.52b). The question now is how and if  $x_{\omega_0}(t)$  converges to  $x(t)$  as  $\omega_0 \rightarrow \infty$ . First, we can ignore the smooth part  $h_{\omega_0} * x_s$ ; it will converge nicely as its Fourier transform decays rapidly. The problematic part is the reconstruction of  $u(t)$  from its restriction  $u_{\omega_0}(t)$ . From the convolution property (3.64)

$$\begin{aligned} u_{\omega_0}(t) &= \int_{-\infty}^{\infty} u(\tau) h_{\omega_0}(t - \tau) d\tau \stackrel{(a)}{=} \sqrt{\frac{\omega_0}{2\pi}} \int_0^{\infty} \text{sinc}(\omega_0(t - \tau)/2) d\tau \\ &\stackrel{(b)}{=} \sqrt{\frac{2}{\omega_0\pi}} \int_{-\infty}^{\omega_0 t/2} \text{sinc}(\tau') d\tau', \end{aligned}$$

where (a) follows from (3.52b), and (b) by change of variable  $\tau' = \omega_0(t - \tau)/2$ .

Recall the shape of the  $\text{sinc}(\tau)$  function in Figure 3.5(b); it has a maximum at  $\tau = 0$ , and zero crossings at multiples of  $\pi$  (except zero). Consider now the above integral. When  $\omega_0 t/2 \rightarrow -\infty$ , the integral vanishes, while as  $\omega_0 t/2 \rightarrow \infty$ , it goes to 1. In between, it oscillates, with local maxima as  $\text{sinc}(\tau)$  changes sign, that is, when  $\tau = k\pi$ , or  $\tau = k\pi/\omega_0$ . The largest oscillations are right around the origin, where  $\text{sinc}(\tau)$  has the largest changes (see Figure 3.7). This overshoot is roughly 9% for the local maxima right next to the origin, decaying as  $O(1/|\tau|)$  farther from the origin.

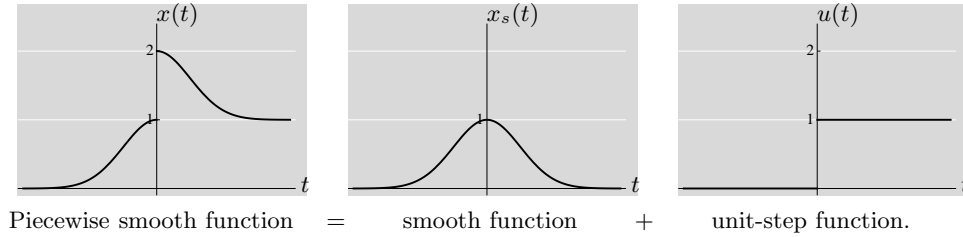
An interesting twist is that the height of the overshoot does not depend on  $\omega_0$ , but only on its location. As  $\omega_0 \rightarrow \infty$ , even though the time axis gets compressed, the same overshoot and undershoot remain. Thus, the maximum error due to these oscillations stays constant regardless of how large  $\omega_0$  gets. This is one of fundamental problems in Fourier analysis of discontinuous functions.

## 3.4. Fourier Transform

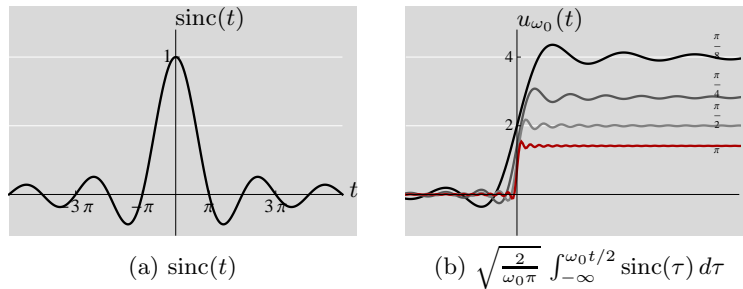
341

FT properties	Time domain	FT domain
<b>Basic properties</b>		
Linearity	$\alpha x(t) + \beta y(t)$	$\alpha X(\omega) + \beta Y(\omega)$
Shift in time	$x(t - t_0)$	$e^{-j\omega t_0} X(\omega)$
Shift in frequency	$e^{j\omega_0 t} x(t)$	$X(\omega - \omega_0)$
Scaling in time and frequency	$x(\alpha t)$	$(1/\alpha) X(\omega/\alpha)$
Time reversal	$x(-t)$	$X(-\omega)$
Differentiation in time	$d^n x(t)/dt^n$	$(j\omega)^n X(\omega)$
Differentiation in frequency	$(-jt)^n x(t)$	$d^n X(\omega)/d\omega^n$
Integration	$\int_{-\infty}^t x(\tau) d\tau$	$X(\omega)/j\omega, X(0) = 0$
Moments	$m_k = \int_{-\infty}^{\infty} t^k x(t) dt = (j)^k d^k X(\omega)/d\omega^k \big _{\omega=0}$	
Convolution in time	$(h * x)(t)$	$H(\omega) X(\omega)$
Convolution in frequency	$h(t) x(t)$	$(1/2\pi)(H * X)(\omega)$
Deterministic autocorrelation	$a(t) = \int_{-\infty}^{\infty} x(\tau) x^*(\tau - t) d\tau$	$A(\omega) =  X(\omega) ^2$
Deterministic crosscorrelation	$c(t) = \int_{-\infty}^{\infty} x(\tau) y^*(\tau - t) d\tau$	$C(\omega) = X(\omega) Y^*(\omega)$
Parseval's equality	$\ x\ ^2 = \int_{-\infty}^{\infty}  x(t) ^2 dt = (1/2\pi) \int_{-\infty}^{\infty}  X(\omega) ^2 d\omega = (1/2\pi) \ X\ ^2$	
<b>Symmetries</b>		
Conjugate	$x^*(t)$	$X^*(-\omega)$
Conjugate, time reversed	$x^*(-t)$	$X^*(\omega)$
Real part	$\Re(x(t))$	$(X(\omega) + X^*(-\omega))/2$
Imaginary part	$\Im(x(t))$	$(X(\omega) - X^*(-\omega))/2j$
Conjugate-symmetric part	$(x(t) + x^*(-t))/2$	$\Re(X(\omega))$
Conjugate-antisymmetric part	$(x(t) - x^*(-t))/2j$	$\Im(X(\omega))$
<b>Symmetries for real <math>x</math></b>		
$X$ conjugate symmetric		$X(\omega) = X^*(-\omega)$
Real part of $X$ even		$\Re(X(\omega)) = \Re(X(-\omega))$
Imaginary part of $X$ odd		$\Im(X(\omega)) = -\Im(X(-\omega))$
Magnitude of $X$ even		$ X(\omega)  =  X(-\omega) $
Phase of $X$ odd		$\arg X(\omega) = -\arg X(-\omega)$
<b>Common transform pairs</b>		
Dirac delta function	$\delta(t)$	1
Shift in time	$\delta(t - t_0)$	$e^{-j\omega t_0}$
Dirac delta comb	$\sum_{n \in \mathbb{Z}} \delta(t - nT)$	$(2\pi/T) \sum_{k \in \mathbb{Z}} \delta(\omega - 2\pi/Tk)$
Exponential function	$e^{-\alpha t }$ $e^{-\alpha t^2}$	$(2\alpha)/(\omega^2 + \alpha^2)$ $\sqrt{\pi/\alpha} e^{-\omega^2/\alpha}$
Ideal lowpass filter	$\sqrt{\frac{\omega_0}{2\pi}} \text{sinc}(\omega_0 t/2)$	$\begin{cases} \sqrt{2\pi/\omega_0}, &  \omega  \leq \omega_0/2; \\ 0, & \text{otherwise.} \end{cases}$
Box function	$\begin{cases} 1/\sqrt{t_0}, &  t  \leq t_0/2; \\ 0, & \text{otherwise,} \end{cases}$	$\sqrt{t_0} \text{sinc}(\omega t_0/2)$
Hat function	$\begin{cases} 1 -  t , &  t  < 1; \\ 0, & \text{otherwise.} \end{cases}$	$(\text{sinc}(\omega/2))^2$

Table 3.2: Properties of the Fourier transform.



**Figure 3.6:** Decomposition of a piecewise smooth function with a single discontinuity into a smooth function and a step function.



**Figure 3.7:** The sinc function and its integral for different values of  $\omega_0$ .

### 3.4.3 Properties of the Fourier Transform

#### Basic Properties

We list here the basic properties of the Fourier transform (assuming it exists); Table 3.2 summarizes these, together with symmetries as well as standard transform pairs.

**Linearity** The Fourier transform operator  $F$  is a linear operator, or,

$$\alpha x(t) + \beta y(t) \xrightarrow{\text{FT}} \alpha X(\omega) + \beta Y(\omega). \quad (3.55)$$

**Shift in Time** The Fourier-transform pair corresponding to a shift in time by  $t_0$  is

$$x(t - t_0) \xrightarrow{\text{FT}} e^{-j\omega t_0} X(\omega). \quad (3.56)$$

**Shift in Frequency** The Fourier-transform pair corresponding to a shift in frequency by  $\omega_0$  is

$$e^{j\omega_0 t} x(t) \xrightarrow{\text{FT}} X(\omega - \omega_0). \quad (3.57)$$

As in Chapter 2, a shift in frequency is often referred to as *modulation* in time, and is dual to the shift in time.

**Scaling in Time and Frequency** The Fourier-transform pair corresponding to scaling in time by  $\alpha$  is scaling in frequency by  $1/\alpha$ ,

$$x(\alpha t) \xleftrightarrow{\text{FT}} \frac{1}{\alpha} X\left(\frac{\omega}{\alpha}\right). \quad (3.58a)$$

We often use normalized rescaling, namely, for  $\alpha > 0$ ,

$$\sqrt{\alpha} x(\alpha t) \xleftrightarrow{\text{FT}} \frac{1}{\sqrt{\alpha}} X\left(\frac{\omega}{\alpha}\right), \quad (3.58b)$$

which conserves the  $\mathcal{L}^2$  norm of  $x(t)$  (and thus of  $X(\omega)$ ), since

$$\|\sqrt{\alpha} x(\alpha t)\|^2 = \int_{-\infty}^{\infty} \alpha |x(\alpha t)|^2 dt \stackrel{(a)}{=} \alpha \int_{-\infty}^{\infty} |x(\tau)|^2 \frac{d\tau}{\alpha} = \|x\|^2, \quad (3.58c)$$

where (a) follows from the change of variable  $\tau = \alpha t$ . This is another of the key properties of the Fourier transform, where a stretch in time is a compaction in frequency, and vice versa.

**Time Reversal** The Fourier-transform pair corresponding to a time reversal  $x(-t)$  is

$$x(-t) \xleftrightarrow{\text{FT}} X(-\omega). \quad (3.59)$$

For a real  $x(t)$ , the Fourier transform of the time-reversed version  $x(-t)$  is  $X^*(\omega)$ .

**Differentiation in Time** The Fourier-transform pair corresponding to differentiation in time is

$$\frac{d^n x(t)}{dt^n} \xleftrightarrow{\text{FT}} (j\omega)^n X(\omega), \quad (3.60)$$

assuming the derivatives exist and are bounded (see Exercise 3.3), or, equivalently, assuming  $\omega^n X(\omega)$  is absolutely integrable.

**Differentiation in Frequency** The Fourier-transform pair corresponding to differentiation in frequency is

$$(-jt)^n x(t) \xleftrightarrow{\text{FT}} \frac{d^n X(\omega)}{d\omega^n}. \quad (3.61a)$$

This is obtained by multiple applications of

$$\int_{-\infty}^{\infty} tx(t)e^{-j\omega t} dt = j \frac{d}{d\omega} \left[ \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt \right] = j \frac{dX(\omega)}{d\omega}, \quad (3.61b)$$

assuming  $t^n x(t)$  for all  $k = 0, 1, \dots, n$  is integrable, and using

$$j \frac{de^{-j\omega t}}{d\omega} = te^{-j\omega t}.$$

This result is dual to differentiation in time above.

**Integration** The Fourier-transform pair corresponding to integration in time is (with  $X(0) = 0$ )

$$\int_{-\infty}^t x(\tau) d\tau \xleftrightarrow{\text{FT}} \frac{1}{j\omega} X(\omega). \quad (3.62)$$

**Moments** Computing the  $k$ th moment using the Fourier transform is

$$m_k = \int_{-\infty}^{\infty} t^k x(t) dt = (j)^k \frac{d^k X(\omega)}{d\omega^k} \Big|_{\omega=0} \quad k \in \mathbb{N}, \quad (3.63a)$$

as a direct application of (3.61a). We give below the first two moments:

$$m_0 = \int_{-\infty}^{\infty} x(t) dt = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \Big|_{\omega=0} = X(0), \quad (3.63b)$$

$$m_1 = \int_{-\infty}^{\infty} tx(t) dt = \int_{-\infty}^{\infty} tx(t) e^{-j\omega t} dt \Big|_{\omega=0} = j \frac{dX(\omega)}{d\omega} \Big|_{\omega=0}. \quad (3.63c)$$

**Convolution in Time** The Fourier-transform pair corresponding to convolution in time is

$$(h * x)(t) \xleftrightarrow{\text{FT}} H(\omega)X(\omega). \quad (3.64)$$

Thus, as in Chapter 2, given a function  $x$  and a filter  $h$ , in the Fourier domain, their convolution maps to the product of the spectrum of the function and frequency response of the filter. This result is a direct consequence of the eigenfunction property of complex exponential functions  $v_\omega$  from (3.45): since each spectral component is the projection of the function  $x$  onto the appropriate invariant space, the Fourier transform diagonalizes the convolution operator. Assume that both  $x$  and  $h$  are in  $\mathcal{L}^1(\mathbb{R})$ . Then,  $h * x$  is also in  $\mathcal{L}^1(\mathbb{R})$ , since

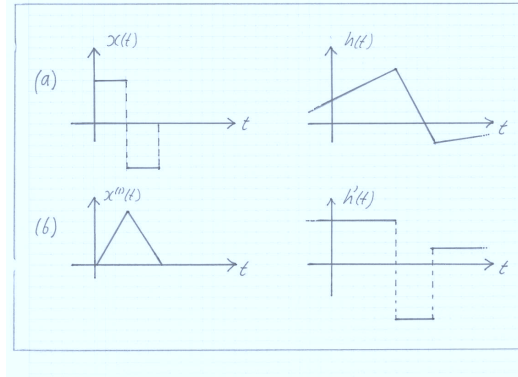
$$\begin{aligned} \int_{-\infty}^{\infty} |(h * x)(t)| dt &= \int_{-\infty}^{\infty} \left| \int_{-\infty}^{\infty} x(\tau) h(t - \tau) d\tau \right| dt \\ &\leq \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |x(\tau)| |h(t - \tau)| d\tau dt \\ &\stackrel{(a)}{=} \int_{-\infty}^{\infty} |x(\tau)| \left( \int_{-\infty}^{\infty} |h(t - \tau)| dt \right) d\tau = \|x\|_1 \|h\|_1, \end{aligned}$$

where (a) follows from Fubini's theorem (see Appendix 1.A.3) allowing for the exchange of the order of integration (allowed since both  $x$  and  $h$  are in  $\mathcal{L}^1(\mathbb{R})$ ).

The spectrum  $Y(\omega)$  of the output function  $y = h * x$  can be written as

$$\begin{aligned} Y(\omega) &= \int_{-\infty}^{\infty} y(t) e^{-j\omega t} dt = \int_{-\infty}^{\infty} \left( \int_{-\infty}^{\infty} x(\tau) h(t - \tau) d\tau \right) e^{-j\omega t} dt \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(\tau) e^{-j\omega\tau} h(t - \tau) e^{-j\omega(t - \tau)} d\tau dt \\ &\stackrel{(a)}{=} \int_{-\infty}^{\infty} x(\tau) e^{-j\omega\tau} d\tau \int_{-\infty}^{\infty} h(t - \tau) e^{-j\omega(t - \tau)} d(t - \tau) = X(\omega)H(\omega), \end{aligned}$$





**Figure 3.8:** Computation of convolution and its equivalent using primitive and derivative instead. (a)  $h * x$ . (b)  $dh(t)/dt * x^{(1)}$ .

where (a) follows again from Fubini's theorem (see Appendix 1.A.3) allowing for the exchange of the order of integration.

While the convolution property states that convolution in time equals to multiplication frequency, the proof does not really convey the intuition we saw in (3.47), which we now repeat. If the function  $x(t)$  has a frequency component at frequency  $\omega_0$ ,  $X(\omega_0)$ , then this frequency will go through the convolution operator without frequency change (the eigenfunction property) but with a weighting given by the frequency response of the filter  $h(t)$  at frequency  $\omega_0$ ,  $H(\omega_0)$  (eigenvalue of the eigenfunction of frequency  $\omega_0$ ). Therefore, the output at frequency  $\omega_0$  is indeed

$$Y(\omega_0) = H(\omega_0)X(\omega_0).$$

**Convolution in Frequency** The Fourier-transform pair corresponding to convolution in frequency is

$$h(t)x(t) \xleftrightarrow{\text{FT}} \frac{1}{2\pi} (H * X)(\omega), \quad (3.65)$$

as to be expected by the duality of time and frequency.

**EXAMPLE 3.7 (DERIVATIVE OF A CONVOLUTION)** Consider  $y = h * x$  and compute its derivative (which we assume to be well defined),

$$\frac{dy(t)}{dt} = \frac{d(h * x)(t)}{dt}. \quad (3.66a)$$

The Fourier transform of  $dy(t)/dt$ , according to (3.60), is

$$\frac{dY(\omega)}{d\omega} = j\omega H(\omega)X(\omega). \quad (3.66b)$$

Since the  $j\omega$  term can be associated either with  $X(\omega)$  or  $H(\omega)$ , using (3.60) with  $n = 1$ , we see that  $dy(t)/dt$  can be written either of the following two ways:

$$\frac{dy(t)}{dt} = h * \frac{dx(t)}{dt} = \frac{dh(t)}{dt} * x. \quad (3.66c)$$

This formula, which resembles integration by parts, is important because it allows one to rewrite any convolution  $h * x$  as the convolution of the primitive of one function and the derivative of the other,

$$h * x = \frac{dh(t)}{dt} * x^{(1)} = h^{(1)} * \frac{dx(t)}{dt}, \quad (3.66d)$$

where  $x^{(1)}$  and  $h^{(1)}$  denote the primitives of  $x(t)$  and  $h(t)$ , respectively. A pictorial example is shown in Figure 3.8.

**Deterministic Autocorrelation** The Fourier-transform pair corresponding to the deterministic autocorrelation of a function  $x(t)$  is

$$a(t) = \int_{-\infty}^{\infty} x(\tau) x^*(\tau - t) d\tau \xleftrightarrow{\text{FT}} A(\omega) = |X(\omega)|^2. \quad (3.67)$$

To show this, express the deterministic autocorrelation as a convolution of  $x$  and its time-reversed version as in (3.37d),  $x(t) * x^*(-t)$ . We know from Table 3.2 that the Fourier transform of  $x^*(-t)$  is  $X^*(\omega)$ . Then, using the convolution property (3.64), we obtain (3.67).

**Deterministic Crosscorrelation** The Fourier-transform pair corresponding to the deterministic crosscorrelation of functions  $x(t)$  and  $y(t)$  is

$$c(t) = \int_{-\infty}^{\infty} x(\tau) y^*(\tau - t) d\tau \xleftrightarrow{\text{FT}} C(\omega) = X(\omega) Y^*(\omega). \quad (3.68)$$

**Parseval's Equality** The Fourier-transform operator  $F$  is a unitary operator (within scaling) and thus preserves the Euclidean norm (see (1.51)),

$$\|x\|^2 = \frac{1}{2\pi} \|Fx\|^2. \quad (3.69a)$$

This follows from

$$\|x\|^2 = \int_{-\infty}^{\infty} |x(t)|^2 dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} |X(\omega)|^2 d\omega = \frac{1}{2\pi} \|X\|^2 = \frac{1}{2\pi} \|Fx\|^2.$$

We now prove Parseval's equality for functions that are both in  $\mathcal{L}^1$  and  $\mathcal{L}^2$ :

$$\begin{aligned} \|x\|^2 &\stackrel{(a)}{=} a(0) = \frac{1}{2\pi} \int_{-\infty}^{\infty} A(\omega) e^{j\omega t} d\omega \Big|_{t=0} \\ &\stackrel{(b)}{=} \frac{1}{2\pi} \int_{-\infty}^{\infty} |X(\omega)|^2 d\omega = \frac{1}{2\pi} \|X(\omega)\|^2, \end{aligned}$$

where (a) follows from (3.14); and (b) from (3.67).

Similarly, *generalized Parseval's equality* can be written as

$$\langle x, y \rangle_t = \frac{1}{2\pi} \langle X, Y \rangle_\omega, \quad (3.69b)$$

following from

$$\langle x, y \rangle_t = \int_{-\infty}^{\infty} x(t) y^*(t) dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) Y^*(\omega) d\omega = \frac{1}{2\pi} \langle X, Y \rangle_\omega.$$

The proof, a simple extension of (3.69a), is left as Exercise 3.4. The key notion emerging from both Parseval's and generalized Parseval's equalities, is the conservation of the  $\mathcal{L}^2$  norm and inner product between time and frequency, showing the Fourier transform to be a unitary map between these domains (up to scaling).

### Transform Pairs

Rather than an exhaustive list of Fourier-transform pairs, we highlight a few which are routinely used, even those that are neither in  $\mathcal{L}^1$  nor in  $\mathcal{L}^2$ . These, as well as some other ones, are summarized in Table 3.2.

**Dirac Delta Function** We summarized some of the commonly used Fourier-transform pairs involving the Dirac delta function in Table 3.2. Note that by now, we are far from  $\mathcal{L}^1$  or  $\mathcal{L}^2$ , since the function or spectrum has no decay, and exist in a generalized sense using the distribution  $\delta(t)$  or  $\delta(\omega)$ . Using this powerful formalism combined with the various properties derived previously, it is easy to compute a plethora of Fourier transforms, for example, the transforms of trigonometric functions, the step function, etc.; see Table 3.2.

Using the shift in time property of the Dirac delta function from Table 3.2 and linearity, we can compute the Fourier transform of the Dirac delta comb (3.7):

$$S_T(\omega) = \sum_{n \in \mathbb{Z}} e^{-j\omega nT}. \quad (3.70)$$

One can show that the above sum is itself a sequence of Dirac delta functions,

$$\sum_{n \in \mathbb{Z}} e^{-j\omega nT} = \frac{2\pi}{T} \sum_{k \in \mathbb{Z}} \delta\left(\omega - \frac{2\pi}{T}k\right). \quad (3.71)$$

This is a direct consequence of the Poisson sum formula, which we discuss next.

**THEOREM 3.11 (POISSON SUM FORMULA)** Given is the Dirac delta comb function (3.7) and a function  $x(t)$  of sufficient decay so that its periodized version

$$x_T(t) = (s_T * x)(t) = \sum_{n \in \mathbb{Z}} x(t - nT) \quad (3.72)$$

converges uniformly. Then:

(i) The Dirac delta comb function and its Fourier-transform pair are:

$$s_T(t) = \sum_{n \in \mathbb{Z}} \delta(t - nT) \xleftrightarrow{\text{FT}} S_T(\omega) = \frac{2\pi}{T} \sum_{k \in \mathbb{Z}} \delta\left(\omega - \frac{2\pi}{T}k\right). \quad (3.73a)$$

(ii) The Poisson sum formula states that:

$$x_T(t) = \sum_{n \in \mathbb{Z}} x(t - nT) = \frac{1}{T} \sum_{k \in \mathbb{Z}} X\left(\frac{2\pi}{T}k\right) e^{j(2\pi/T)kt}. \quad (3.73b)$$

(iii) As a corollary, for  $T = 1$  and  $t = 0$ ,

$$\sum_{n \in \mathbb{Z}} x(n) = \sum_{n \in \mathbb{Z}} x_n = \sum_{k \in \mathbb{Z}} X(2\pi k). \quad (3.73c)$$

(iv) The following is a Fourier-transform pair:

$$\langle x(t), x(t - n) \rangle_t = \delta_n \xleftrightarrow{\text{FT}} \sum_{k \in \mathbb{Z}} |X(\omega + 2\pi k)|^2 = 1. \quad (3.73d)$$

*Proof.* The proof of (3.73a) is somewhat technical and is left to Exercise 3.5. Instead, we use it to show (3.73b).

Since  $x_T(t) = s_T * x$ , we can use the convolution property (3.64) to obtain its Fourier-transform pair as

$$\begin{aligned} X_T(\omega) &\stackrel{(a)}{=} S_T(\omega) X(\omega) \stackrel{(b)}{=} \frac{2\pi}{T} \sum_{k \in \mathbb{Z}} \delta\left(\omega - \frac{2\pi}{T}k\right) X(\omega) \\ &\stackrel{(c)}{=} \frac{2\pi}{T} \sum_{k \in \mathbb{Z}} X\left(\frac{2\pi}{T}k\right) \delta\left(\omega - \frac{2\pi}{T}k\right), \end{aligned} \quad (3.74)$$

where (a) follows from the convolution property (3.64); (b) from (3.73a), and (c) from the multiplication property in Table 3.1. Taking the inverse Fourier transform of (3.74), we get

$$\begin{aligned} x_T(t) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} X_T(\omega) e^{j\omega t} d\omega = \frac{1}{T} \int_{-\infty}^{\infty} \sum_{k \in \mathbb{Z}} X\left(\frac{2\pi}{T}k\right) \delta\left(\omega - \frac{2\pi}{T}k\right) e^{j\omega t} d\omega \\ &= \frac{1}{T} \sum_{k \in \mathbb{Z}} X\left(\frac{2\pi}{T}k\right) \int_{-\infty}^{\infty} \delta\left(\omega - \frac{2\pi}{T}k\right) e^{j\omega t} d\omega \\ &\stackrel{(a)}{=} \frac{1}{T} \sum_{k \in \mathbb{Z}} X\left(\frac{2\pi}{T}k\right) e^{j(2\pi/T)kt}, \end{aligned}$$

where (a) follows again from the multiplication property in Table 3.1.

To prove (3.73d), observe that the left-hand side is the deterministic autocorre-

## 3.4. Fourier Transform

349

lation  $a(\tau)$  evaluated at integers  $\tau = n$ ,

$$\begin{aligned}\delta_n &= \langle x(t), x(t-n) \rangle_t = \int_{-\infty}^{\infty} x(t)x(t-n) dt \stackrel{(a)}{=} \int_{-\infty}^{\infty} x(t)x(t-\tau) \sum_{n \in \mathbb{Z}} \delta(\tau-n) dt, \\ &\stackrel{(b)}{=} \int_{-\infty}^{\infty} x(t)x(t-\tau) dt \sum_{n \in \mathbb{Z}} \delta(\tau-n) \stackrel{(c)}{=} a(\tau) s_1(\tau),\end{aligned}$$

where (a) follows from Table 3.1 and the fact that the expression is nonzero only for  $n = 0$ ; (b) from pulling the Dirac delta comb out of the integral; and (c) from the definition of the deterministic autocorrelation (3.14) (we assumed  $x(t)$  was real) and Dirac delta comb for  $T = 1$  (3.7). Taking the Fourier transform of the above, we get

$$\begin{aligned}1 &\stackrel{(a)}{=} \frac{1}{2\pi} A(\omega) * S_1(\omega) \stackrel{(b)}{=} \frac{1}{2\pi} |X(\omega)|^2 * 2\pi \sum_{k \in \mathbb{Z}} \delta(\omega - 2\pi k) \\ &\stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} |X(\omega)|^2 * \delta(\omega - 2\pi k) \stackrel{(d)}{=} \sum_{k \in \mathbb{Z}} |X(\omega - 2\pi k)|^2,\end{aligned}$$

where (a) follows from the convolution-in-frequency property of the Fourier transform (3.65); (b) from (3.67) and (3.73a); in (c) we pulled out the summation in front of the convolution; and (d) from Table 3.1.

The Poisson sum formula has many applications; in signal processing, it is most often used in the proof of the sampling theorem (see Chapter 4), since the process of sampling can be described as a multiplication of an input signal  $x(t)$  by a Dirac delta comb  $s_T(t)$ .

**Sinc Function** While we have seen the Fourier-transform pair of the sinc function a few times so far, we repeat it here for completeness. The sinc function was given in (2.8a) and Example 3.5. The sinc Fourier-transform pair is (scaled so that the time-domain function is unit norm):

$$x(t) = \sqrt{\frac{\omega_0}{2\pi}} \text{sinc}(\omega_0 t/2) \xleftrightarrow{\text{FT}} X(\omega) = \begin{cases} \sqrt{2\pi/\omega_0}, & |\omega| \leq \omega_0/2; \\ 0, & \text{otherwise.} \end{cases} \quad (3.75)$$

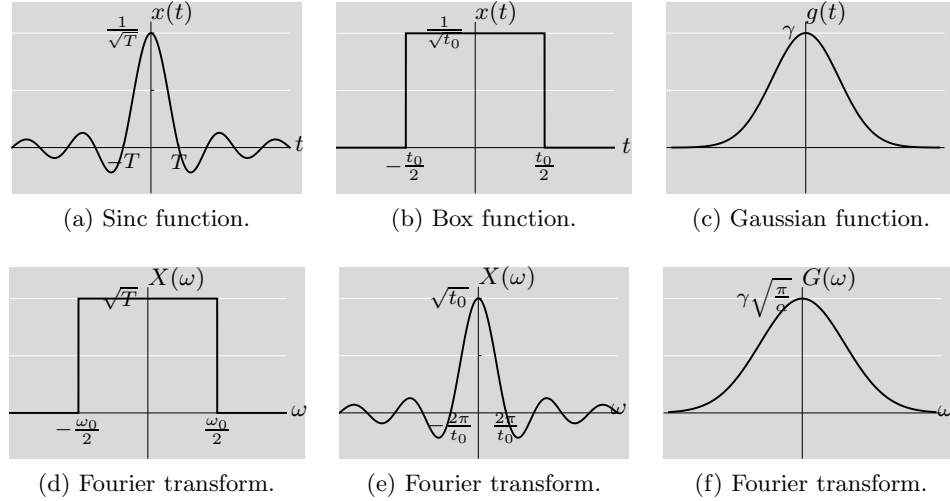
In other words, the Fourier transform of a sinc is a box function in frequency, or, a lowpass filter (see Figure 3.9(a) and (d)).

**Box Function** By duality, the Fourier transform of the box function (3.11) in time is the sinc function in frequency (see Figure 3.9(b) and (e)):

$$x(t) = \begin{cases} 1/\sqrt{t_0}, & |t| \leq t_0/2; \\ 0, & \text{otherwise,} \end{cases} \xleftrightarrow{\text{FT}} X(\omega) = \sqrt{t_0} \text{sinc}(\omega t_0/2). \quad (3.76)$$

**Heaviside Function** The Heaviside function was defined in (3.8). Its Fourier transform (in the sense of distributions) is

$$u(t) = \begin{cases} 1, & t \geq 0; \\ 0, & \text{otherwise.} \end{cases} \xleftrightarrow{\text{FT}} U(\omega) = \frac{1}{2} \delta(\omega) - \frac{1}{2j\omega}. \quad (3.77)$$

**Figure 3.9:** Time and frequency views of a function.

**Gaussian Function** The Fourier transform of a Gaussian function in time is a Gaussian function in frequency (see Figure 3.9(c) and (f)):

$$g(t) = \gamma e^{-\alpha t^2} \xleftrightarrow{\text{FT}} G(\omega) = \gamma \sqrt{\frac{\pi}{\alpha}} e^{-\omega^2/(4\alpha)}, \quad (3.78)$$

for  $\alpha, \gamma$  positive real constants. That this is a Fourier-transform pair can be proven in various ways; the easiest is to observe that  $e^{-\alpha t^2}$  is the solution of the differential equation

$$\frac{dx(t)}{dt} + 2\alpha x(t) = 0.$$

Taking the Fourier transform and using (3.61b) and (3.60), leads to an equivalent differential equation in Fourier domain, which, when solved, yields (3.78) (the topic of Exercise 3.7).

### Decay and Smoothness

The Fourier transform allows us to go from the time domain to its dual, the frequency domain, and back. Given this duality and the nature of the Fourier transform, which takes a global view, local properties can be mapped to global features, and vice versa. Our goal is to characterize functions in terms of how smooth, or, *regular*, they are. As we will see, the decay of the corresponding Fourier transform is a powerful indicator of such regularity. We start with the coarsest version: functions with  $p$  continuous derivatives. We follow with Lipschitz regularity, which allows for a more local measurement on an interval or even at a point.

**$C^p$  Regularity** A global view the Fourier transform allows for easy characterization of functions with  $p$  continuous derivatives, that is, those functions belonging to  $C^p$  spaces (see Section 1.2). We have seen one such result in Section 3.4.2 (and Solved Exercise 3.3), where we stated that if  $x(t) \in \mathcal{L}^1(\mathbb{R})$ , then its Fourier transform  $X(\omega)$  is bounded and continuous. Conversely, if  $|X(\omega)|$  decays faster than  $1/|\omega|$  for large  $|\omega|$ , then  $x(t)$  is bounded and continuous. More precisely, if

$$|X(\omega)| \leq \frac{\gamma}{1 + |\omega|^{1+\varepsilon}}, \quad (3.79a)$$

for some constant  $\gamma$  and  $\varepsilon > 0$ , then  $|X(\omega)|$  is absolutely integrable, or,  $X(\omega) \in \mathcal{L}^1$ . Therefore, the inverse Fourier transform  $x(t)$  is bounded and continuous, or,  $x(t) \in C^0$ . We can easily extend this argument (Exercise 3.9) to show that if

$$|X(\omega)| \leq \frac{\gamma}{1 + |\omega|^{p+1+\varepsilon}} \quad (3.79b)$$

then  $x(t) \in C^p$ . Conversely, for  $x(t) \in C^p$ , its Fourier transform is bounded by

$$|X(\omega)| \leq \frac{1}{1 + |\omega|^{p+1}}. \quad (3.79c)$$

For  $x(t) \in C^0$  ( $p = 0$ ), its Fourier transform is bounded by

$$|X(\omega)| \leq \frac{1}{1 + |\omega|}. \quad (3.79d)$$

The slight asymmetry between (3.79b) and (3.79c) comes from the fact that if  $|X(\omega)|$  decays as  $1/(1 + |\omega|^{p+1})$ , then it can happen that the  $p$ th derivative exists but might be discontinuous, as we show now.

**EXAMPLE 3.8 (DECAY AND SMOOTHNESS)** Let us consider a few examples.

- (i) We start with the box function (3.11) (with  $t_0 = 1$ ) and its Fourier transform  $X(\omega)$  (3.51a). Since  $X(\omega)$  is a sinc, the best we can do is to say that it is bounded by  $\gamma/(1 + |\omega|)$ . Therefore, we cannot say whether  $x(t)$  is continuous (of course, we know it is not).
- (ii) Next, consider the hat function  $x(t)$  from (3.49a). It is continuous, and thus, its Fourier transform decays at least as  $\gamma/(1 + |\omega|)$ , for some  $\gamma$ , according to (3.79d); actually, it decays much faster, as  $\gamma/(1 + |\omega|^2)$ .<sup>74</sup>
- (iii) Finally, we look at a function that has fast decay and is very smooth except for a single point  $t = 0$  where its derivative is discontinuous:

$$x(t) = e^{-\alpha|t|}. \quad (3.80a)$$

<sup>74</sup>Note that throughout we use a generic constant  $\gamma$  with the understanding that each  $\gamma$  is different from the previous one. This is to avoid introducing several constants.

It is thus a  $C^0$  function, even though it is  $C^\infty$  everywhere except at the origin. Its Fourier transform is

$$\begin{aligned} X(\omega) &= \int_{-\infty}^{\infty} e^{-\alpha|t|} e^{-j\omega t} dt \stackrel{(a)}{=} \int_0^{\infty} e^{-(\alpha-j\omega)t} dt + \int_0^{\infty} e^{-(\alpha+j\omega)t} dt \\ &= \frac{1}{\alpha + j\omega} + \frac{1}{\alpha - j\omega} = \frac{2\alpha}{(\alpha^2 + \omega^2)}, \end{aligned} \quad (3.80b)$$

where in (a) we split the integral over negative/positive  $t$ , respectively, and then changed variable in the first one (sign change). Since  $x(t)$  is continuous (that is, it is in  $C^0$ ), according to (3.79d) its Fourier transform must decay at least as  $\gamma/(1 + |\omega|)$ . Conversely, since the Fourier transform decays exactly as  $\gamma/(1 + |\omega|^2)$  but not faster, we cannot say more than that  $x(t)$  must be continuous.

These three examples illustrate the power of the Fourier transform in analyzing functions; we will see in the latter part of the book that, in addition, wavelet tools allow for local characterization as well.

**Lipschitz Regularity** Membership in a  $C^p$  class gives us a coarse idea of the smoothness of a function. A finer analysis, allowing for characterizing local smoothness of a function on an interval or even a point, is possible using Lipschitz (or Hölder) exponents. Beware, however, that the Fourier characterization which we discuss here, is again global as before.

**DEFINITION 3.12 (LIPSCHITZ REGULARITY)** A function  $x(t)$  is called Lipschitz of order  $\alpha$ ,  $0 \leq \alpha < 1$ , when, for any  $t$  and  $t_0$ ,

$$|x(t) - x(t_0)| \leq c|t - t_0|^\alpha. \quad (3.81)$$

Higher-order characterization for  $r = n + \alpha$ ,  $n \in \mathbb{N}$ , can be obtained by replacing  $x(t)$  by its  $n$ th derivative. Note that Lipschitz regularity provides a sense of noninteger differentiability, but is actually weaker than the differentiability of the same integer order. For example, the hat function (3.49a) is Lipschitz of order  $(1 - \varepsilon)$  for any positive  $\varepsilon$ , while only  $C^0$ .

To see how this regularity manifests itself in the Fourier domain, similarly to (3.79b), we show in Solved Exercise 3.10 that a function  $x(t)$  is bounded and Lipschitz  $\alpha$  when

$$\int_{-\infty}^{\infty} |X(\omega)|(1 + |\omega|^\alpha) d\omega < \infty, \quad (3.82)$$

providing a global characterization of regularity. Interestingly, Lipschitz regularity can be defined locally, thus providing a local characterization as well; we will have to wait on the wavelet transform in Part II to see such local characterization.



### 3.4.4 Frequency Response of Filters

The Fourier transform is defined for functions and we use spectrum to denote their Fourier transforms. The frequency response is defined for filters (systems) as

$$H(\omega) = \int_{-\infty}^{\infty} h(t)e^{-j\omega t} dt, \quad \omega \in \mathbb{R} \quad (3.83a)$$

with the corresponding impulse response,

$$h(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} H(\omega)e^{j\omega t} d\omega, \quad t \in \mathbb{R}. \quad (3.83b)$$

As in discrete time, we often write the magnitude and phase of the frequency response separately:

$$H(\omega) = |H(\omega)| e^{j \arg(H(\omega))},$$

where  $|H(\omega)|$  is a real positive function—the *magnitude response*, while  $\arg(H(\omega))$  is a real function between  $-\pi$  and  $\pi$ —the *phase response*.

**Diagonalization of the Convolution Operator** As in Chapter 2, from the form of (3.48a), the Fourier transform is clearly a linear operator. Let us denote this through  $X = Fx$ . The adjoint  $F^*$  is also a linear operator, and  $(1/2\pi)F^*$  is the inverse Fourier-transform operator. Thus  $(1/2\pi)F^*F$  is an identity operator.

While the Fourier-transform operator  $F$  diagonalizes any convolution operator  $H$ , this is a bit subtle. Though we cannot describe  $H$  as having a matrix representation in a basis associated with the Fourier transform, the essence of diagonalization is present in (3.47). The composition  $FHF^*$  is a linear operator on the space of functions. Since the input  $X(\omega)$  can be expanded as  $X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt$ , we can conclude from (3.47) that  $Y = (1/2\pi)FHF^*X$  results in  $Y(\omega) = H(\omega)X(\omega)$ . This captures the notion of *diagonalization* without writing a diagonal matrix.

### 3.4.5 Laplace Transform

In the previous chapter, we introduced the  $z$ -transform, which extends the DTFT from the unit circle to the complex plane, by constructing spaces spanned by complex exponentials of nonunit magnitude. The  $z$ -transform is used when DTFT does not exist. Similarly, in this chapter, we introduce the Laplace transform, which extends the Fourier transform from the exponential of the imaginary axis to that of the entire complex plane, by constructing spaces spanned by general complex exponentials. The Laplace transform is used when the Fourier transform does not exist. The two transforms have similar properties, allowing us to extend the validity of many results to a larger class of functions.

**DEFINITION 3.13 (LAPLACE TRANSFORM)** The Laplace transform of a function  $x(t)$  is a function of  $s = \sigma + j\omega$ ,  $s \in \mathbb{C}$  given by

$$X(s) = \int_{-\infty}^{\infty} x(t)e^{-st} dt. \quad (3.84)$$

When  $s = j\omega$ , the Laplace transform is simply the Fourier transform; when  $s = \sigma + j\omega$ , the Laplace transform is the Fourier transform of  $x'(t) = x(t)e^{-\sigma t}$ , or

$$X(\sigma + j\omega) = \int_{-\infty}^{\infty} \underbrace{x(t)e^{-\sigma t}}_{x'(t)} e^{-j\omega t} dt. \quad (3.85)$$

Therefore, convergence of the Laplace transform depends solely on the exponent  $\sigma$ , and the ROC of the integral (3.84) consists of vertical strips. In particular, when  $x(t)e^{-\sigma t}$  is absolutely integrable, then (3.84) converges pointwise to a bounded and continuous function. Just as for the  $z$ -transform, the Laplace transform and its associated ROC define the time-domain function. The Laplace transform satisfies a number of properties which follow directly from their Fourier counterparts; selected ones are summarized in Table 3.3. We now illustrate the necessity of associating an ROC to a Laplace transform of a function.

LT properties	Time domain	LT domain	ROC
Linearity	$\alpha x(t) + \beta y(t)$	$\alpha X(s) + \beta Y(s)$	$\supset \text{ROC}_x \cap \text{ROC}_y$
Shift in time	$x(t - t_0)$	$e^{-st_0} X(s)$	$\text{ROC}_x$
Shift in $s$	$e^{St} x(t)$	$X(s - S)$	$\text{ROC}_x + S$
Scaling in time	$x(\alpha t)$	$(1/\alpha) X(s/\alpha)$	$\alpha \text{ROC}_x$
Convolution in time	$(h * x)(t)$	$H(s) X(s)$	$\supset \text{ROC}_h \cap \text{ROC}_x$

**Table 3.3:** Selected properties of the Laplace transform.

**EXAMPLE 3.9 (LAPLACE TRANSFORM OF THE HEAVISIDE FUNCTION)** Consider the Heaviside function  $x(t) = u(t)$  defined in (3.8). Its Fourier transform exists in the sense of distributions, and is given in (3.77). Its Laplace transform

$$X(s) = \int_{-\infty}^{\infty} x(t)e^{-st} dt = \int_0^{\infty} e^{-st} dt, \quad (3.86a)$$

is well defined for  $\sigma > 0$ , yielding  $X(s) = 1/s$ :

$$x(t) \xleftrightarrow{\text{LT}} X(s) = \frac{1}{s}, \quad \text{ROC} = \{s \mid \Re(s) > 0\}. \quad (3.86b)$$

However,  $x(t) = -u(-t)$  does not have a Fourier transform (this is a bit technical), but the Laplace transform exists and is also  $X(s) = 1/s$ , for  $\sigma < 0$ :

$$x(t) \xleftrightarrow{\text{LT}} X(s) = \frac{1}{s}, \quad \text{ROC} = \{s \mid \Re(s) < 0\}. \quad (3.86c)$$

This example shows two important points. (1) The Laplace transform must have an ROC associated with it, since otherwise, two different functions can have the same Laplace transform expression. (2) Functions  $x(t)$  for which the Fourier transform does not exist, can have a well-defined Laplace transform.

## 3.5 Fourier Series

Periodic functions, as well as finite-length functions circularly extended, as introduced in Section 3.2, have a series expansion in terms of Fourier coefficients. They are of great mathematical interest as questions of convergence occupied mathematicians for a long time after Fourier's original work. Moreover, the Fourier series is just the dual of the discrete-time Fourier transform seen in the previous chapter, and thus, its study is central to discrete-time signal processing as well.

As we have done so far, we will also look at the Fourier series in terms of orthonormal bases and their geometrical properties, and will thus be interested in functions having a square integrable period. For such functions, the central result is the completeness of the Fourier series basis.

Our treatment of the Fourier series will be brief; most of its properties are similar to those of the Fourier transform, or, its discrete-time counterpart, DFT. We follow a similar path as before, and show how, by defining an appropriate convolution for periodic functions, the Fourier series emerges naturally. We then define the Fourier series formally and proceed to discuss a few of its properties, including the duality with the DTFT.

### 3.5.1 Definition of the Fourier Series

**Eigenfunctions of the Circular Convolution Operator** We now follow the same path from both the previous chapter and the discussion on Fourier transform in Section 3.4. Clearly, our aim is to find spaces of functions that are invariant under the operation of circular convolution we defined in (3.9). In other words, we want to find eigenfunctions  $v$  of the circular convolution operator  $H$  as in (3.42).

These eigenfunctions are clearly going to be some form of a complex exponential function; however, since we are dealing with the periodic convolution operator, these will be of the form (compare this to (3.44) as well as (2.156) in Chapter 2):

$$v_k(t) = e^{j\omega_0 kt} = e^{j(2\pi/T)kt}, \quad (3.87)$$

generating the corresponding spaces  $S_k = \{\alpha e^{j(2\pi/T)kt} \mid \alpha \in \mathbb{C}, k \in \mathbb{Z}\}$ . Because they have to be periodic with period  $T$ , there are countably many of these spaces (index  $k \in \mathbb{Z}$ ) as opposed to uncountably many for the Fourier transform (index  $\omega \in \mathbb{R}$ ). The quantity  $k$  is called *discrete frequency*.

We can now follow exactly the same process to check what happens if we apply the circular convolution operator onto one of these eigensfunctions:

$$\begin{aligned} H v_k(t) &= h \circledast v_k(t) = \int_{-T/2}^{T/2} v_k(t - \tau) h(\tau) d\tau = \int_{-T/2}^{T/2} e^{j(2\pi/T)k(t-\tau)} h(\tau) d\tau \\ &= \underbrace{\int_{-T/2}^{T/2} h(\tau) e^{-j(2\pi/T)k\tau} d\tau}_{\lambda_k = H_k} \underbrace{e^{j(2\pi/T)kt}}_{v_k(t)}. \end{aligned} \quad (3.88)$$

As expected, applying the circular convolution operator to the complex exponential function  $v_k(t) = e^{j(2\pi/T)kt}$  results in the same function, albeit scaled by the corre-

sponding eigenvalue  $\lambda_k$ , we call, as before, the *frequency response*  $H_k$  of the system. We can thus rewrite (3.88) as

$$H e^{j(2\pi/T)kt} = h \circledast e^{j(2\pi/T)kt} = H_k e^{j(2\pi/T)kt}. \quad (3.89)$$

**Fourier Series** Finding the appropriate Fourier transform of  $x$  now amounts to projecting  $x$  onto each of the  $S_k$ :

**DEFINITION 3.14 (FOURIER SERIES)** The Fourier series of a periodic function  $x(t)$  with period  $T$  is

$$X_k = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-j(2\pi/T)kt} dt, \quad k \in \mathbb{Z}; \quad (3.90a)$$

we call it the *spectrum* of  $x$ . The inverse Fourier series of  $X_k$  is

$$x(t) = \sum_{k \in \mathbb{Z}} X_k e^{j(2\pi/T)kt}, \quad t \in [-T/2, T/2). \quad (3.90b)$$

It exists when (3.90b) converges for all  $t$ . When the inverse Fourier series exists, we denote the Fourier-series pair:

$$x(t) \xleftrightarrow{\text{FS}} X_k.$$

One often denotes  $\omega_0 = 2\pi/T$  as the fundamental frequency  $1/T$  expressed in radians. The coefficients  $X_k$  are called *Fourier coefficients*. The factor  $1/T$  in (3.90a) ensures that the transform is unitary as we will see shortly. Exercise 3.11 explores the connection between the real Fourier series and (3.90b).

**Fourier Series as an Orthonormal Basis** When a period of a function  $x(t)$  is square integrable, the inversion of the Fourier series in the  $\mathcal{L}^2$  sense is guaranteed. The best way to see this is by considering the complex exponentials of frequency  $\{(2\pi/T)k\}_{k \in \mathbb{Z}}$  as an orthonormal basis for the interval  $[-T/2, T/2)$ .

**THEOREM 3.15 (FOURIER SERIES AS AN ORTHONORMAL BASIS)** The set

$$\varphi_k(t) = \frac{1}{\sqrt{T}} e^{j(2\pi/T)kt}, \quad k \in \mathbb{Z}, \quad t \in [-T/2, T/2), \quad (3.91)$$

forms an orthonormal basis for  $\mathcal{L}^2([-T/2, T/2))$ .

*Proof.* It is easy to see that  $\{\varphi_k(t)\}$  is an orthonormal system, since

$$\langle \varphi_k, \varphi_\ell \rangle = \frac{1}{T} \int_{-T/2}^{T/2} e^{j(2\pi/T)(k-\ell)t} dt = \delta_{k-\ell}. \quad (3.92)$$

To show completeness, compute the normalized Fourier series coefficients of a function  $x(t)$  in  $\mathcal{L}^2([-T/2, T/2])$ :

$$X_k = \langle x, \varphi_k \rangle = \frac{1}{\sqrt{T}} \int_{-T/2}^{T/2} x(t) e^{-j(2\pi/T)kt} dt, \quad (3.93)$$

and the  $(2N+1)$ -term approximation of  $x(t)$ ,

$$\hat{x}_N(t) = \frac{1}{\sqrt{T}} \sum_{k=-N}^N X_k e^{j(2\pi/T)kt}. \quad (3.94)$$

This approximation is an orthogonal projection of  $x(t)$  onto the subspace spanned by  $\{\varphi_k(t)\}_{k=-N}^N$ . We now need to show that

$$\lim_{N \rightarrow \infty} \int_{-T/2}^{T/2} |x(t) - \hat{x}_N(t)|^2 dt = 0, \quad (3.95)$$

for any  $x(t)$  in  $\mathcal{L}^2([-T/2, T/2])$ . This is left to Exercise 3.12, which considers continuous functions. The result can also be extended to general  $\mathcal{L}^2$  functions by the argument that continuous functions are dense in  $\mathcal{L}^2$ .

This Hilbert space view of Fourier series is not only important from a mathematical point of view, but also leads to geometric intuition, for example, on least squares approximation. We summarize this Hilbert space view in the following theorem:

**THEOREM 3.16 (FOURIER SERIES IN  $\mathcal{L}^2$ )** Given a  $T$ -periodic  $x(t) \in \mathcal{L}^2([-T/2, T/2])$ :

(i)  *$\mathcal{L}^2$  inversion*: The function  $x(t)$  can be written as

$$x(t) = \sum_{k \in \mathbb{Z}} X_k e^{j(2\pi/T)kt} \quad (3.96a)$$

with Fourier coefficients

$$X_k = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-j(2\pi/T)kt} dt, \quad (3.96b)$$

with equality in the  $\mathcal{L}^2$  sense.

(ii) *Norm conservation*: The map is an isometry (distance-preserving map) between the interval  $[-T/2, T/2]$  and  $\ell^2(\mathbb{Z})$  (up to a scale factor) given by the generalized Parseval's equality

$$\langle x, y \rangle_t = \int_{-T/2}^{T/2} x(t) y^*(t) dt = T \sum_{k \in \mathbb{Z}} X_k Y_k^*, \quad (3.97a)$$

or, for  $x(t) = y(t)$ , the Parseval's equality

$$\|x\|^2 = \int_{-T/2}^{T/2} |x(t)|^2 dt = T \sum_{k \in \mathbb{Z}} |X_k|^2. \quad (3.97b)$$

(iii) *Best least squares approximation:* The function

$$\hat{x}_N(t) = \sum_{k=-N}^N X_k e^{j(2\pi/T)kt} \quad (3.97c)$$

is the best least squares approximation of  $x(t)$  on the subspace  $S$  spanned by  $\{\varphi_k(t)\}_{k=-N}^N$ .

*Proof.* Part (i) follows from the orthonormal basis property of  $\{\varphi_k(t)\}_{k \in \mathbb{Z}}$  shown in the previous theorem, and Part (ii) is equivalent to (P3.12-2) with renormalization. Finally, consider an arbitrary function  $y_N(t) \in S$ ,

$$y_N(t) = \sum_{k=-N}^N \alpha_k e^{-j(2\pi/T)kt}.$$

Now, following (3.97b),

$$\frac{1}{T} \int_{-T/2}^{T/2} |x(t) - y_N(t)|^2 dt = \sum_{k \in \mathbb{Z}} |X_k - Y_k|^2 = \sum_{k=-N}^N |X_k - \alpha_k|^2 + \sum_{|k| > N} |X_k|^2,$$

which is minimized when  $\alpha_k = X_k$  for  $k \in [-N, N]$ , or  $y_N(t) = \hat{x}_N(t)$ .

**Relation of the Fourier Series to the Fourier Transform** What is the relation of the Fourier series coefficients of a periodic function  $x(t)$  to the Fourier transform of one period of that same function? Call that one period  $I = [-T/2, T/2)$ , and  $x_I(t) = x(t)\chi_{[-T/2, T/2)}(t)$  the restriction of  $x(t)$  to  $I$ , that is,  $x_I$  is equal to  $x(t)$  on  $I$  and is 0 otherwise, it is easy to verify (Solved Exercise 3.4) that

$$X_k = \frac{1}{T} X_I \left( \frac{2\pi}{T} k \right), \quad (3.98)$$

where  $X_I$  is the Fourier transform of  $x_I$ . In other words, the Fourier series coefficients of  $x(t)$  are samples of the Fourier transform of the same function restricted to one period  $I = [-T/2, T/2)$ .

**Relation of the Fourier Series to the DTFT** Consider the Fourier series of a  $2\pi$ -periodic function, or, from (3.90a)

$$X_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} x(t) e^{-jkt} dt. \quad (3.99)$$

We can recognize this as the inverse DTFT in (2.78b). In other words, the inverse DTFT expresses the sequence  $x_n$  as the Fourier series of a  $2\pi$ -periodic DTFT  $X(e^{j\omega})$ . Conversely, a periodic function  $x(t)$  can be seen as the DTFT of the Fourier series sequence  $X_k$ . Table 3.5 summarizes this duality that helps us understand concepts in one domain, such as, for example, convergence and orthogonality, through the same concepts in the other.

### 3.5.2 Existence of the Fourier Series

When the Fourier series coefficients of a  $T$ -periodic function are absolutely summable, then, just as in the Fourier-transform case, we have an inversion formula.

**THEOREM 3.17 ( $\mathcal{L}^1$  INVERSION)** Given  $T$ -periodic  $x(t)$  with one period in  $\mathcal{L}^1(\mathbb{R})$  and Fourier series coefficients  $X_k \in \ell^1(\mathbb{Z})$ , then for almost all  $t$ ,

$$x(t) = \sum_{k \in \mathbb{Z}} X_k e^{jk\omega_0 t}, \quad (3.100)$$

where  $\omega_0 = 2\pi/T$ . In addition, if  $x(t)$  is continuous, then equality holds everywhere.

The proof is similar to the equivalent proof for the Fourier transform. From the inversion formula, the following uniqueness holds: if two functions have the same period  $T$  and the same Fourier coefficients, then they are equal almost everywhere (Exercise 3.13).

### 3.5.3 Properties of the Fourier Series

#### Basic Properties

We list here the basic properties of the Fourier series (assuming it exists); Table 3.4 summarizes these, together with symmetries as well as standard transform pairs.

**Linearity** The Fourier series operator  $F$  is a linear operator, or,

$$\alpha x(t) + \beta y(t) \xleftrightarrow{\text{FS}} \alpha X_k + \beta Y_k. \quad (3.101)$$

**Shift in Time** The Fourier-series pair corresponding to a shift in time by  $t_0$  is

$$x(t - t_0) \xleftrightarrow{\text{FS}} e^{-j(2\pi/T)kt_0} X_k. \quad (3.102)$$

**Shift in Frequency** The Fourier-series pair corresponding to a shift in frequency by  $k_0$  is

$$e^{j(2\pi/T)k_0 t} x(t) \xleftrightarrow{\text{FS}} X_{k-k_0}. \quad (3.103)$$

As in Chapter 2, a shift in frequency is often referred to as *modulation* in time, and is dual to the shift in time.

**Time Reversal** The Fourier-series pair corresponding to a time reversal  $x(-t)$  is

$$x(-t) \xleftrightarrow{\text{FS}} X_{-k}. \quad (3.104)$$

FS properties	Time domain	FS domain
<b>Basic properties</b>		
Linearity	$\alpha x(t) + \beta y(t)$	$\alpha X_k + \beta Y_k$
Shift in time	$x(t - t_0)$	$e^{-j(2\pi/T)kt_0} X_k$
Shift in frequency	$e^{j(2\pi/T)k_0 t} x(t)$	$X(k - k_0)$
Time reversal	$x(-t)$	$X_{-k}$
Differentiation	$d^n x(t)/dt^n$	$(j2\pi k/T)^n X_k$
Integration	$\int_{-T/2}^t x(\tau) d\tau$	$(T/j2\pi k) X_k, X_0 = 0$
Convolution in time	$(h \otimes x)(t)$	$T H_k X_k$
Convolution in frequency	$h(t) x(t)$	$(H \otimes X)_k$
Deterministic autocorrelation	$a(t) = \int_{-T/2}^{T/2} x(\tau) x^*(\tau - t) d\tau$	$A_k = T  X_k ^2$
Deterministic crosscorrelation	$c(t) = \int_{-T/2}^{T/2} x(\tau) y^*(\tau - t) d\tau$	$C_k = T X_k Y_k^*$
Parseval's equality	$\ x\ ^2 = \int_{-T/2}^{T/2}  x(t) ^2 dt = T \sum_{k \in \mathbb{Z}}  X_k ^2 = T \ X\ ^2$	
<b>Symmetries</b>		
Conjugate	$x^*(t)$	$X_{-k}^*$
Conjugate, time reversed	$x^*(-t)$	$X_k^*$
Real part	$\Re(x(t))$	$(X_k + X_{-k}^*)/2$
Imaginary part	$\Im(x(t))$	$(X_k - X_{-k}^*)/2j$
Conjugate-symmetric part	$(x(t) + x^*(-t))/2$	$\Re(X_k)$
Conjugate-antisymmetric part	$(x(t) - x^*(-t))/2j$	$\Im(X_k)$
<b>Symmetries for real <math>x</math></b>		
$X$ conjugate symmetric		$X_k = X_{-k}^*$
Real part of $X$ even		$\Re(X_k) = \Re(X_{-k})$
Imaginary part of $X$ odd		$\Im(X_k) = -\Im(X_{-k})$
Magnitude of $X$ even		$ X_k  =  X_{-k} $
Phase of $X$ odd		$\arg X_k = -\arg X_{-k}$
<b>Common transform pairs</b>		
Ideal lowpass filter	$\sqrt{\frac{k_0}{T}} \frac{\text{sinc}(\pi k_0 t/T)}{\text{sinc}(\pi t/T)}$	$\begin{cases} 1/\sqrt{k_0 T}, &  k  \leq (k_0 - 1)/2; \\ 0, & \text{otherwise.} \end{cases}$
Box function	$\begin{cases} 1/\sqrt{t_0}, &  t  \leq t_0/2; \\ 0, & \text{otherwise,} \end{cases}$	$\frac{\sqrt{t_0}}{T} \text{sinc}(\pi k t_0/T)$

**Table 3.4:** Properties of the Fourier series.

**Differentiation** The Fourier-series pair corresponding to differentiation in time is

$$\frac{d^n x(t)}{dt^n} \xleftrightarrow{\text{FS}} (j \frac{2\pi}{T} k)^n X_k. \quad (3.105)$$



## 3.5. Fourier Series

361

This can be derived as follows:

$$\begin{aligned} \int_{-T/2}^{T/2} x'(t) e^{-j(2\pi/T)kt} dt &= x(t) e^{-j(2\pi/T)kt} \Big|_{-T/2}^{T/2} - \int_{-T/2}^{T/2} x(t) \left(-j \frac{2\pi}{T} k\right) e^{-j(2\pi/T)kt} dt \\ &= j \frac{2\pi}{T} k \int_{-T/2}^{T/2} x(t) e^{-j(2\pi/T)kt} dt = j \frac{2\pi}{T} k X_k. \end{aligned}$$

A formula for the primitive of  $x(t)$ , assuming it has zero mean ( $X_0 = 0$ ), can be derived similarly (Exercise 3.14).

**Integration** The Fourier-series pair corresponding to integration in time is (with  $X_0 = 0$ )

$$\int_{-T/2}^t x(\tau) d\tau \xleftrightarrow{\text{FS}} \frac{T}{j2\pi k} X_k. \quad (3.106)$$

**Convolution in Time** As expected, given a periodic function  $x$  and a filter  $h$  (not necessarily periodic), the convolution property states that, in the Fourier domain, their convolution maps to the product of the Fourier coefficients of the function and frequency response of the filter:

$$(h \otimes x)(t) \xleftrightarrow{\text{FS}} T H_k X_k. \quad (3.107)$$

Exercise 3.15 proves the property when the filter  $h$  is a periodized version of an infinite-length filter, and uses the fact that the frequency response of  $h$  is obtained by sampling the frequency response of that infinite-length filter.

**Convolution in Frequency** The Fourier-series pair corresponding to convolution in frequency is

$$h(t) x(t) \xleftrightarrow{\text{FS}} (H \otimes X)_k, \quad (3.108)$$

as to be expected by the duality of time and frequency.

**Deterministic Autocorrelation** The Fourier-series pair corresponding to the deterministic autocorrelation of a function  $x(t)$  is

$$a(t) = \int_{-T/2}^{T/2} x(\tau) x^*(\tau - t) d\tau \xleftrightarrow{\text{FS}} A_k = T |X_k|^2. \quad (3.109)$$

**Deterministic Crosscorrelation** The Fourier-series pair corresponding to the deterministic crosscorrelation of functions  $x(t)$  and  $y(t)$  is

$$c(t) = \int_{-T/2}^{T/2} x(\tau) y^*(\tau - t) d\tau \xleftrightarrow{\text{FS}} C_k = T X_k Y_k^*. \quad (3.110)$$

**Parseval's Equality** The Fourier-transform operator  $F$  is a unitary operator (within scaling) and thus preserves the Euclidean norm (see (1.51)):

$$\|x\|^2 = T \|Fx\|^2. \quad (3.111)$$

This follows from

$$= \int_{-T/2}^{T/2} |x(t)|^2 dt = T \sum_{k \in \mathbb{Z}} |X_k|^2 = T \|X\|^2 = T \|Fx\|^2.$$

### Transform Pairs

We consider a couple of simple periodic functions to illustrate the behavior of Fourier series by applying some of the properties seen above.

**Square Wave** Consider a square wave of period  $T = 1$ , with one period given by

$$x(t) = \begin{cases} 1, & -1/2 \leq t < 0; \\ -1, & 0 \leq t < 1/2. \end{cases} \quad (3.112)$$

This function is real and antisymmetric; its Fourier series coefficients are thus purely imaginary (see Table 3.5):

$$\begin{aligned} X_k &= \int_{-1/2}^0 e^{-j2\pi kt} dt - \int_0^{1/2} e^{-j2\pi kt} dt \\ &= \frac{1}{j2\pi k} (\cos(\pi k) - 1) = \frac{1}{j2\pi k} ((-1)^k - 1) \\ &= \begin{cases} 2j/\pi k, & k \text{ odd}; \\ 0, & k \text{ even}. \end{cases} \end{aligned} \quad (3.113)$$

**Triangle Wave** Consider a triangle wave of period  $T = 1$ , with one period given by

$$y(t) = \int_{-1/2}^t x(\tau) d\tau = \begin{cases} 1/2 - |t|, & |t| \leq 1/2. \end{cases} \quad (3.114)$$

We can use the integral property (3.106) to find

$$Y_k = \begin{cases} 1/4, & k = 0; \\ 0, & k \text{ even}, k \neq 0; \\ 1/(\pi k)^2, & k \text{ odd}. \end{cases} \quad (3.115)$$

Exercise 3.16 explores alternate ways to compute this Fourier series.

### Decay and Smoothness

Similarly to the Fourier transform, smoothness of the periodic function and decay of its Fourier coefficients are related, as seen, for example, in (3.98), which relates

## 3.6. Continuous Stochastic Processes and Systems

363

the Fourier transform of one period with the Fourier coefficients. In (3.113), we saw that a discontinuous function such as the square wave has Fourier coefficients decaying only as  $O(1/k)$  as, while a continuous function such as the triangle leads to a decay of  $O(1/k^2)$ .

More generally, using the derivative property, one can show that if

$$\sum_{k \in \mathbb{Z}} |k|^p |X_k| < \infty, \quad (3.116)$$

then  $x(t)$  is  $p$  times differentiable. For example, the triangle wave satisfies the above for  $p = 0$  and is thus continuous. Exercise 3.17 explores the sawtooth function and the Gibbs phenomenon.

## 3.5.4 Frequency Response of Filters

The frequency response of a filter and the inverse frequency response are given by:

$$H_k = \frac{1}{T} \int_{-T/2}^{T/2} h(t) e^{-j(2\pi/T)kt} dt, \quad k \in \mathbb{Z}, \quad (3.117a)$$

$$h(t) = \sum_{k \in \mathbb{Z}} H_k e^{j(2\pi/T)kt}, \quad t \in \mathbb{R}. \quad (3.117b)$$

**Diagonalization of the Circular Convolution Operator** Again, from (3.89) we can immediately see that the Fourier series operator  $F$  in  $X = Fx$  diagonalizes the convolution operator  $H$  as in (3.42). As this is a central concept, we show it here explicitly. We start with the output of the circular convolution of  $h$  and  $x$ :

$$\begin{aligned} y(t) &\stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} Y_k e^{j(2\pi/T)kt} \stackrel{(b)}{=} \sum_{k \in \mathbb{Z}} H_k X_k e^{j(2\pi/T)kt} \\ &\stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} H_k \left( \frac{1}{T} \int_{-T/2}^{T/2} x(\tau) e^{-j(2\pi/T)k\tau} d\tau \right) e^{j(2\pi/T)kt}, \end{aligned}$$

where (a) follows from the definition of the inverse Fourier series, (3.90b); (b) from the convolution-in-time property (3.64); and (c) from the definition of the Fourier series, (3.90a). Figure 3.10 illustrates this diagonalization property.

## 3.6 Continuous Stochastic Processes and Systems

As in Chapter 2, we now consider processes and systems in the presence of uncertainty. We use standard probability theory introduced in Section 1.C to model uncertainty. To introduce the stochastic framework, this section follows the same structure as does the chapter: We start with continuous stochastic processes (random processes), followed by systems (almost exclusively LSI systems) and associated functions both in time domain as averages in the form of means, variances and correlation functions, as well as in frequency domain such as power spectral density.

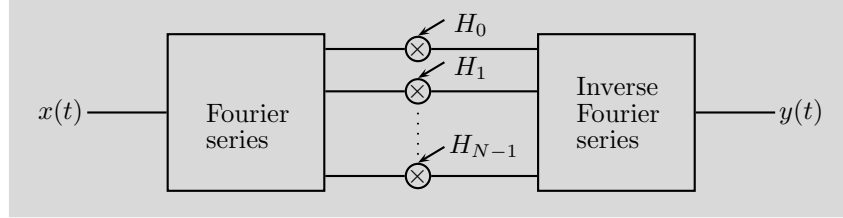


Figure 3.10: Diagonalization property of the Fourier series.

### 3.6.1 Processes

A continuous stochastic process is a function supported on  $\mathbb{R}$ , characterized by specifying the PDF for all tuples  $(t_1, t_2, \dots, t_N)$ ,  $N$  arbitrary; in other words, it is a random process (see Section 1.C). If all random variables have the same distribution and are independent, the process is called *i.i.d.* (*independent and identically distributed*). We use the following functions defined on a continuous stochastic process:<sup>75</sup>

mean	$\mu_x(t)$	$E[x(t)];$	(3.118)
variance	$\text{var}(x(t))$	$E[(x(t) - \mu_x(t))^2];$	
standard deviation	$\sigma_x(t)$	$\sqrt{\text{var}(x(t))};$	
autocorrelation	$a_x(\tau, t)$	$E[x(\tau) x^*(\tau - t)].$	

Most of the time we will assume we are dealing with continuous *wide-sense stationary* (WSS) processes, that is, those whose mean is constant and autocorrelation depends only on  $t$ :

$$\mu_x(t) = \mu_x; \quad a_x(\tau, t) = a_x(t). \quad (3.119)$$

Often, this assumption is valid at least for a portion of time of a given process.

**White Noise** A very particular continuous stochastic process appearing widely in signal processing is the *white noise*<sup>76</sup> process  $x$ , whose mean is zero and its autocorrelation is  $a(t) = \sigma_x^2 \delta(t)$ , or,

$$\mu_x(t) = 0; \quad \text{var}(x(t)) = \sigma_x^2; \quad \sigma_x(t) = \sigma_x; \quad a_x(t) = \sigma_x^2 \delta(t). \quad (3.120)$$

If the underlying PDF is Gaussian,  $x$  is called *white Gaussian noise*, or sometimes, *additive white Gaussian noise* (AWGN).

The white noise process is uncorrelated, but not always independent (in the case of the Gaussian PDF, it will automatically be independent as well). Often, the term *whitening*, or, *decorrelation* is used, meaning that a given process is made to have zero mean and Dirac delta function-like autocorrelation function, and is basically a diagonalization process for the covariance matrix.

<sup>75</sup>Although we have seen a version of these in Section 1.C, we repeat them here for completeness.

<sup>76</sup>As in Chapter 2, the Fourier transform of the autocorrelation of white noise is a constant, mimicking the behavior of the spectrum of the white light; thus the term white noise.

### 3.6.2 Systems

We now assume that our input  $x$  is a continuous WSS process, and the system is LSI described by its impulse response  $h$ , as in Section 2.3.3. Note that the system is deterministic, given by a fixed impulse response  $h$ . What can we say about the output? It is given by (2.58), and we can compute the same functions on the output we computed on the input (mean, variance, standard deviation and autocorrelation). We start with the mean:

$$\begin{aligned}\mu_y(t) &= E[y(t)] \stackrel{(a)}{=} E\left[\int_{-\infty}^{\infty} x(\tau) h(t-\tau) d\tau\right] \stackrel{(b)}{=} \int_{-\infty}^{\infty} E[x(\tau)] h(t-\tau) d\tau \\ &\stackrel{(c)}{=} \mu_x \int_{-\infty}^{\infty} h(t-\tau) d\tau \stackrel{(d)}{=} \mu_x H(e^{j0}) = \mu_y,\end{aligned}\quad (3.121a)$$

where (a) follows from (3.36); (b) from the linearity of the expectation; (c) from  $x$  being WSS; and (d) from the frequency response of the LSI system. We can thus see that the mean of the output is a constant, independent of  $t$ . The variance is

$$\text{var}(y(t)) = a_{y,0} - (\mu_y)^2, \quad (3.121b)$$

and the autocorrelation of the output

$$\begin{aligned}a_y(\tau, t) &= E[y(\tau) y^*(\tau - t)] \\ &\stackrel{(a)}{=} E\left[\int_{-\infty}^{\infty} x(\tau - q) h(q) dq \int_{-\infty}^{\infty} x^*(\tau - t - r) h^*(r) dr\right] \\ &\stackrel{(b)}{=} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(q) h^*(r) E[x(\tau - q) x^*(\tau - t - r)] dr dq \\ &\stackrel{(c)}{=} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(q) h^*(r) a_x(t - (q - r)) dr dq \\ &\stackrel{(d)}{=} \int_{-\infty}^{\infty} \left( \int_{-\infty}^{\infty} h(q) h^*(q - p) dq \right) a_x(t - p) dp \\ &\stackrel{(e)}{=} \int_{-\infty}^{\infty} a_h(p) a_x(t - p) dp = a_y(t),\end{aligned}\quad (3.121c)$$

where (a) follows from the definition of convolution (3.36); (b) from linearity of expectation; (c) from the expression for the autocorrelation of the WSS  $x$ , (3.118); (d) from change of variable  $p = q - r$ ; and (e) from the definition of deterministic autocorrelation (3.14). From this, we can see that if the input is WSS, the output is WSS as well (as the mean is a constant and the autocorrelation depends only on the difference  $t$ ). We also see that the autocorrelation of the output is the convolution of the stochastic autocorrelation of the input and the deterministic autocorrelation of the impulse response of the system.

Similarly, we compute the crosscorrelation between the input and the output:

$$\begin{aligned}
 c_{x,y}(\tau, t) &= E[x(\tau) y^*(\tau - t)] \\
 &\stackrel{(a)}{=} E[x(\tau) \int_{-\infty}^{\infty} h^*(r) x^*(\tau - t - r) dr] \\
 &= E[\int_{-\infty}^{\infty} h^*(r) x(\tau) x^*(\tau - (t + r)) dr] \\
 &\stackrel{(b)}{=} \int_{-\infty}^{\infty} h^*(r) E[x(\tau) x^*(\tau - (t + r))] dr \\
 &\stackrel{(c)}{=} \int_{-\infty}^{\infty} h^*(r) a_x(t + r) dr, \tag{3.121d}
 \end{aligned}$$

where (a) follows from (3.36); (b) from linearity of expectation; and (c) from the expression for the autocorrelation of the WSS  $x$ , (3.118). We will use the above expressions shortly to make some important observations in the Fourier domain.

### 3.6.3 Fourier Transform

As for deterministic functions, we can use Fourier techniques to gain insight into the behavior of continuous stochastic processes and systems. While we cannot take a Fourier transform of a continuous stochastic process, as it is neither absolutely, nor square integrable, we make assessments based on averages (moments). We can take the Fourier transform of the autocorrelation, and we do this right now. Let us assume a WSS  $x$  and find the Fourier transform of its autocorrelation (3.118) (which we assume to have sufficient decay so as to be absolutely, or, at least square integrable):

$$A_x(\omega) = \int_{-\infty}^{\infty} a_x(t) e^{-j\omega t} dt, \tag{3.122}$$

which is called the *power spectral density*. The power spectral density exists if and only if  $x$  is WSS, the result of the Wiener-Khinchin theorem. When  $x$  is real, the power spectral density is nonnegative, and thus admits a spectral factorization

$$A_x(\omega) = U(\omega) U^*(\omega),$$

where  $U(\omega)$  is its nonunique spectral root.

Given (3.121c), the autocorrelation of the output can be expressed as

$$A_y(\omega) = A_h(\omega) A_x(\omega) = |H(\omega)|^2 A_x(\omega), \tag{3.123}$$

where  $A_h(\omega) = |H(\omega)|^2$  is the Fourier transform of the deterministic autocorrelation of  $h$ , according to Table 3.2. The quantity

$$E[y^2(t)] = a_y(0) = \frac{1}{2\pi} \int |H(\omega)|^2 A_x(\omega) d\omega,$$

is the *average output power*. Similarly to (3.123), and from (3.121d), we can express the crosscorrelation between the input and the output as

$$C_{x,y}(\omega) = H^*(\omega) A_x(\omega). \tag{3.124}$$

---

3.6. Continuous Stochastic Processes and Systems

367

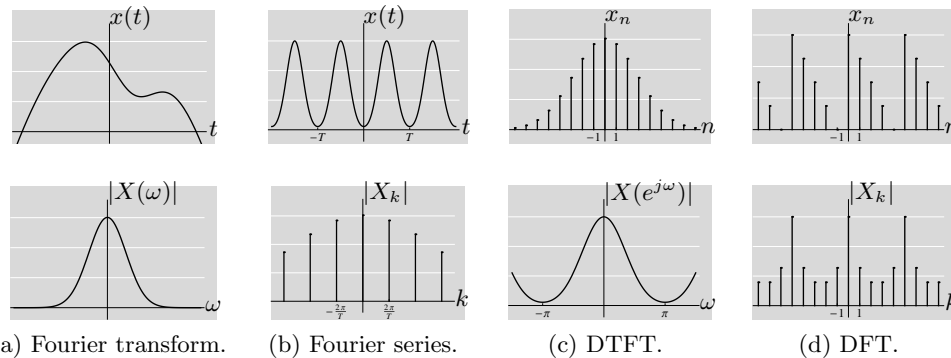
**White Noise** Using (3.120) and Table 3.2, we see that the power spectral density of white noise is a constant:

$$A(\omega) = \sigma_x^2, \quad (3.125)$$

and has infinite power since its autocorrelation is a Dirac delta function.

## Chapter at a Glance

By now, we have seen all the versions of the Fourier transform and series that will be used in the sequel, summarized in Table 3.5 and Figure 3.11. These variants of the Fourier transform differ depending on the underlying space of sequences or functions, and can be continuous infinite (Fourier transform), discrete infinite (Fourier series), continuous finite and circularly extended (DTFT), and discrete finite and circularly extended (DFT).



**Figure 3.11:** Various forms of Fourier transforms seen in this chapter and Chapter 2.

Transform	Forward/Inverse	Duality
Fourier transform	$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt$ $x(t) = (1/2\pi) \int_{-\infty}^{\infty} X(\omega) e^{j\omega t} d\omega$	
Fourier series	$X_k = (1/T) \int_{-T/2}^{T/2} x(t) e^{-j(2\pi/T)kt} dt$ $x(t) = \sum_k X_k e^{j(2\pi/T)kt}$	Dual with DTFT $x(t+T) = x(t)$
Discrete-time Fourier transform	$X(e^{j\omega}) = \sum_n x_n e^{-j\omega n}$ $x_n = (1/2\pi) \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega$	Dual with Fourier series $X(e^{j(\omega+2\pi)}) = X(e^{j\omega})$
Discrete Fourier transform	$X_k = \sum_{n=0}^{N-1} x_n e^{-j(2\pi/N)kn}$ $x_n = (1/N) \sum_{k=0}^{N-1} X_k e^{j(2\pi/N)kn}$	

**Table 3.5:** Various forms of Fourier transforms seen in this chapter and Chapter 2.

In both this chapter as well as the previous one, box and sinc functions played a prominent role; for easy reference, they are summarized in the following table.



---

**Unit-Norm Box and Sinc Functions/Sequences**


---

**Functions on the real line**

$$x(t), \quad t \in \mathbb{R}, \quad \|x\| = 1$$

$$\text{Box} \quad \begin{cases} 1/\sqrt{t_0}, & |t| \leq t_0/2; \\ 0, & \text{otherwise,} \end{cases}$$

$$\text{Sinc} \quad \sqrt{\frac{\omega_0}{2\pi}} \text{sinc}(\omega_0 t/2)$$

**FT**

$$X(\omega), \quad \omega \in \mathbb{R}, \quad \|X\| = \sqrt{2\pi}$$

$$\sqrt{t_0} \text{sinc}(\omega t_0/2)$$

$$\begin{cases} \sqrt{2\pi/\omega_0}, & |\omega| \leq \omega_0/2; \\ 0, & \text{otherwise.} \end{cases}$$

**Periodic functions**

$$x(t), \quad t \in [-T/2, T/2), \quad \|x\| = 1$$

$$\text{Box} \quad \begin{cases} 1/\sqrt{t_0}, & |t| \leq t_0/2; \\ 0, & \text{otherwise,} \end{cases}$$

$$\text{Sinc} \quad \sqrt{\frac{k_0}{T}} \frac{\text{sinc}(\pi k_0 t/T)}{\text{sinc}(\pi t/T)}$$

**FS**

$$X_k, \quad k \in \mathbb{Z}, \quad \|X\| = 1/\sqrt{T}$$

$$\frac{\sqrt{t_0}}{T} \text{sinc}(\pi k t_0/T)$$

$$\begin{cases} 1/\sqrt{k_0 T}, & |k| \leq (k_0 - 1)/2; \\ 0, & \text{otherwise.} \end{cases}$$

**Infinite-length sequences**

$$x_n, \quad n \in \mathbb{Z}, \quad \|x\| = 1$$

$$\text{Box} \quad \begin{cases} 1/\sqrt{n_0}, & |n| \leq (n_0 - 1)/2; \\ 0, & \text{otherwise,} \end{cases}$$

$$\text{Sinc} \quad \sqrt{\frac{\omega_0}{2\pi}} \text{sinc}(\omega_0 n/2)$$

**DTFT**

$$X(e^{j\omega}), \quad \omega \in [-\pi, \pi), \quad \|X\| = \sqrt{2\pi}$$

$$\sqrt{n_0} \frac{\text{sinc}(\omega n_0/2)}{\text{sinc}(\omega/2)}$$

$$\begin{cases} \sqrt{2\pi/\omega_0}, & |\omega| \leq \omega_0/2; \\ 0, & \text{otherwise.} \end{cases}$$

**Finite-length sequences**

$$x_n, \quad n \in \{0, 1, \dots, N-1\}, \quad \|x\| = 1$$

$$\text{Box} \quad \begin{cases} 1/\sqrt{n_0}, & |n - N/2| \geq (n_0 - 1)/2; \\ 0, & \text{otherwise,} \end{cases}$$

$$\text{Sinc} \quad \sqrt{\frac{k_0}{N}} \frac{\text{sinc}(\pi n k_0/N)}{\text{sinc}(\pi n/N)}$$

**DFT**

$$X_k, \quad k \in \{0, 1, \dots, N-1\}, \quad \|X\| = \sqrt{N}$$

$$\sqrt{n_0} \frac{\text{sinc}(\pi n_0 k/N)}{\text{sinc}(\pi k/N)}$$

$$\begin{cases} \sqrt{N/k_0}, & |k - N/2| \geq (k_0 - 1)/2; \\ 0, & \text{otherwise.} \end{cases}$$


---

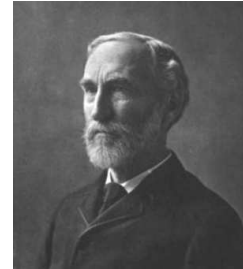
**Table 3.6:** Unit-norm box and sinc functions/sequences seen in this chapter and Chapter 2. The box function/sequence is sometimes termed Dirichlet; thus, the box is the Dirichlet in time and the sinc is the Dirichlet in frequency. Moreover, the box function/sequence is usually used as a rectangular window, while the sinc function/sequence is the impulse response of an ideal lowpass filter. For the DTFT,  $\omega_0 = 2\pi/N$  leads to an ideal  $N$ th-band lowpass filter, while  $\omega_0 = \pi$  leads to an ideal halfband lowpass filter. In the DFT, the inequalities appear reversed to account for the circularity of the DFT.

**Historical Remarks**

A giant featured in the title of this chapter, **Jean Baptiste Joseph Fourier (1768-1830)**, was a French mathematician and physicist, who proposed his famous Fourier series while working on the equations for heat flow. His interests were varied, his biography unusual. He followed Napoleon to Egypt and spent a few years in Cairo, even contributing a few papers to the Egyptian Institute Napoleon founded. He served as a permanent

secretary of the French Academy of Science. He published “Théorie analytique de la chaleur”, in which he claimed that any function can be decomposed into a sum of sines and cosines; while we know that is only partially true, it took mathematicians a long time to tighten the result. Lagrange and Dirichlet both worked on it, with Dirichlet finally formulating conditions under which Fourier series exists.

**Josiah Willard Gibbs (1839-1903)** was an American mathematician, physicist and chemist. Known for many significant contributions (among others as the inventor of vector analysis, independently of Heaviside), he was also the one to remark upon the unusual way the Fourier series behaves at a discontinuity; the Fourier series overshoots significantly, though in a controlled manner. Moreover, we can get into trouble by trying to differentiate the Fourier series. **Karl Theodor Wilhelm Weierstrass (1815-1897)** gave a famous example of a continuous function not differentiable anywhere. In 1926, **Andrey Nikolaevich Kolmogorov (1903-1987)** proved that there existed a function, periodic and locally Lebesgue integrable, with a divergent Fourier series at all points. While this seemed as another strike against Fourier series, **Lennart Carleson (1928)**, a Swedish mathematician, showed in 1966 that every periodic, locally square-integrable function has a Fourier series that converges almost everywhere.



## Further Reading

**Books and Textbooks** We now give a sample list of books/textbooks in which more information can be found about various topics we discussed in this chapter. Some of them are standard in the signal processing community and others we have used while writing this book.

More on the Dirac delta function can be found in [109]. Mallat wrote a book on wavelets and signal processing, similar in outlook to this one, but aimed at applied mathematicians, which has a fair amount of material on harmonic analysis [100]. The book by Brémaud [19] is a clean, self-contained text aimed at the signal processing researcher. The text by Bracewell [17] is a classic; the material is written with engineering in mind, with plenty of intuition. The book by Papoulis [112] is another classic engineering text which has been used for quite some time. The material on stochastics can be found in the book by Porat [113]. Finally, the text by Folland [54] has been written from a physicist's point of view, and offers an excellent treatment of partial differential equations.

## Exercises with Solutions

### 3.1. Derivative of the Dirac Delta Function

Given is a continuously differentiable function  $x(t)$ . Show that

$$\int_{t \in \mathbb{R}} x(t) \delta'(t) dt = -x'(0).$$

where  $\delta'(t)$  is the derivative of the Dirac delta function.

*Solution:* Using integration by parts,

$$\int_{t \in \mathbb{R}} x(t) \delta'(t) dt = x(t) \delta(t) \Big|_{-\infty}^{\infty} - \int_{t \in \mathbb{R}} x'(t) \delta(t) dt \stackrel{(a)}{=} - \int_{t \in \mathbb{R}} x'(t) \delta(t) dt \stackrel{(b)}{=} -x'(0),$$

where (a) follows from  $x(t)\delta(t)$  being zero at  $\pm\infty$ ; and (b) from Table 3.1.

### 3.2. Properties of the Convolution

- (i) Prove the associativity property (3.37c).
- (ii) Derive a counterexample to associativity for some function not in  $\mathcal{L}^1$ .  
(Hint: Use a function such as a Heaviside function.)
- (iii) Prove the deterministic autocorrelation property (3.37d).

*Solution:* TBD

### 3.3. Fourier Transform of Functions in $\mathcal{L}^1(\mathbb{R})$

Given is  $x(t) \in \mathcal{L}^1(\mathbb{R})$ . Show the following properties of its Fourier transform  $X(\omega)$ :

- (i)  $X(\omega)$  is bounded.
- (ii)  $X(\omega)$  is continuous.

(Hint: Use the dominated convergence theorem, (1.197), on  $(e^{-j\omega t} - e^{-j\Omega t})x(t)$ .)

*Solution:*

- (i) To show that  $X(\omega)$  is bounded, we write:

$$|X(\omega)| = \left| \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \right| \stackrel{(a)}{\leq} \int_{-\infty}^{\infty} |x(t) e^{-j\omega t}| dt = \int_{-\infty}^{\infty} |x(t)| dt \stackrel{(b)}{<} \infty,$$

where (a) follows from Section 1.A.4; and (b) from  $x(t) \in \mathcal{L}^1(\mathbb{R})$ .

- (ii) To show that  $X(\omega)$  is continuous, consider  $(e^{-j\omega t} - e^{-j\Omega t})x(t)$ . Now,

$$|(e^{-j\omega t} - e^{-j\Omega t})x(t)| \leq |(e^{-j\omega t} - e^{-j\Omega t})| |x(t)| = 2|x(t)|.$$

We have thus a positive dominating function that is integrable, and we can use the dominated convergence theorem (see Appendix 1.A.3) to allow us to replace the limit of integrals by the integral of the limit, that is,

$$\begin{aligned} \lim_{\omega \rightarrow \Omega} X(\omega) - X(\Omega) &= \lim_{\omega \rightarrow \Omega} \int_{-\infty}^{\infty} (e^{-j\omega t} - e^{-j\Omega t}) x(t) dt \\ &= \int_{-\infty}^{\infty} \lim_{\omega \rightarrow \Omega} (e^{-j\omega t} - e^{-j\Omega t}) x(t) dt = 0, \end{aligned}$$

proving continuity of  $X(\omega)$ .

### 3.4. Relation of the Fourier Series to the Fourier Transform

Verify (3.98), that is, that the Fourier series coefficients of  $x(t)$  are samples of the Fourier transform of the same function restricted to the interval  $I = [-T/2, T/2]$ .

*Solution:* The Fourier series coefficients of a periodic  $x(t)$  are given by (3.90a):

$$X_k = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-j(2\pi/T)kt} dt. \quad (\text{E3.4-1})$$

On the other hand, given a restriction of  $x(t)$  to the interval  $I = [-T/2, T/2]$  ( $x_I$  is equal to  $x(t)$  on  $[-T/2, T/2]$  and is 0 otherwise), its Fourier transform is

$$X_I(\omega) = \int_{t \in \mathbb{R}} x_I(t) e^{-j\omega t} dt = \int_{-T/2}^{T/2} x(t) e^{-j\omega t} dt. \quad (\text{E3.4-2})$$

Comparing (E3.4-2) and (E3.4-1), we see that

$$X_k = \frac{1}{T} X_I\left(\frac{2\pi}{T}k\right). \quad (\text{E3.4-3})$$

### 3.5. Convolution and Sum of Continuous Random Variables

Let  $x$  and  $y$  be independent continuous random variables with PDFs  $f_x$  and  $f_y$ . Show that

$z = x + y$  has PDF  $f_z = f_x * f_y$ .

*Solution:* For any  $t \in \mathbb{R}$ ,

$$\begin{aligned} F_z(t) &= P(z \leq t) = P(x + y \leq t) \stackrel{(a)}{=} \int_{-\infty}^{\infty} \int_{-\infty}^{t-u} f_{x,y}(s, u) ds du \\ &\stackrel{(b)}{=} \int_{-\infty}^{\infty} \int_{-\infty}^{t-u} f_x(s) f_y(u) ds du \stackrel{(c)}{=} \int_{-\infty}^{\infty} f_y(u) \left( \int_{-\infty}^{t-u} f_x(s) ds \right) du \\ &\stackrel{(d)}{=} \int_{-\infty}^{\infty} f_y(u) F_x(t-u) du \end{aligned}$$

where (a) follows from expressing the region  $\{(s, u) \mid s + u \leq t\}$ ; (b) from the independence of  $x$  and  $y$ ; (c) from  $f_y(y)$  not depending on  $s$ ; and (d) from the definition of  $F_x$ . Differentiating with respect to  $t$  gives

$$f_z(t) = \int_{-\infty}^{\infty} f_y(u) f_x(t-u) du,$$

showing that  $f_z = f_x * f_y$ .

## Exercises

### 3.1. Properties of the Dirac Delta Function

Recall the sequence  $d_n$  from (3.6).

- (i) Prove that  $d_1, d_2, \dots$  does not converge in  $\mathcal{L}^2$  norm.  
(*Hint:* Recall from Definition 1.13 that convergence in a vector space depends on the norm. It is adequate to show that  $d_1, d_2, \dots$  is not a Cauchy sequence under the  $\mathcal{L}^2$  norm.)
- (ii) Prove that if  $x(t)$  is continuous at  $t_0$ , then

$$\lim_{n \rightarrow \infty} \int_{-\infty}^{\infty} x(t) d_n(t - t_0) dt = x(t_0)$$

- (iii) Prove that if  $x(t)$  is continuous at  $t_0$ , then

$$\lim_{n \rightarrow \infty} \int_{-\infty}^{\infty} y(t) x(t) d_n(t - t_0) dt = \lim_{n \rightarrow \infty} \int_{-\infty}^{\infty} y(t) x(t_0) d_n(t - t_0) dt$$

for any function  $y(t)$ .

- (iv) Suppose  $x(t)$  is a continuously differentiable function. Show that

$$\int_{-\infty}^{\infty} x(t) \delta'(t) dt = -x'(0)$$

follows from the properties of the Dirac delta function and the assumption that integration by parts is valid for expressions involving  $\delta(t)$  and its derivative. Find similar expressions for higher-order derivatives of  $\delta(t)$ .

### 3.2. BIBO Stability

Given is an LSI differential equation with impulse response  $h(t)$ . The output is given by the convolution integral (3.36)

$$y(t) = \int_{\tau \in \mathbb{R}} x(\tau) h(t - \tau) d\tau.$$

Prove that  $h(t) \in \mathcal{L}^1(\mathbb{R})$  is a necessary and sufficient condition for BIBO stability.

### 3.3. Derivative of a Function

Given is a function  $x(t)$  and its Fourier transform  $X(\omega)$ . Give a sufficient condition on  $X(\omega)$  for the derivative  $\partial x(t)/\partial t$  to be bounded.

## 3.4. Generalized Parseval's Equality

Prove that if  $x(t)$ ,  $y(t)$  are both in  $\mathcal{L}^1 \cap \mathcal{L}^2$ , then

$$\langle x, y \rangle_t = \frac{1}{2\pi} \langle X, Y \rangle_\omega,$$

where  $X(\omega)$ ,  $Y(\omega)$ , are their Fourier transforms, respectively.

(Hint: Express the inner product as a convolution of  $x(t)$  and  $y^*(-t)$  evaluated at the origin, and use the convolution theorem.)

## 3.5. Poisson Sum Formula

Prove (3.71),

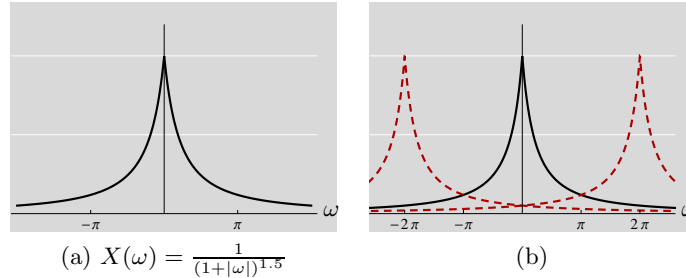
$$S_T(\omega) = \sum_{n \in \mathbb{Z}} e^{-j\omega n T} = \frac{2\pi}{T} \sum_{k \in \mathbb{Z}} \delta\left(\omega - \frac{2\pi}{T} k\right).$$

(Hint: Since  $S_T(\omega)$  is  $2\pi/T$ -periodic, show that on one period,  $S_T(\omega) = (2\pi/T)\delta(\omega)$ . Since this is a distribution, show that for any continuous test function  $X(\omega)$  with support in  $[-\pi/T, \pi/T]$ , the following holds:

$$\lim_{N \rightarrow \infty} \int_{\omega \in \mathbb{R}} \sum_{n=-N}^N e^{-j\omega n T} X(\omega) d\omega = \frac{2\pi}{T} X(0).$$

## 3.6. Application of Poisson Sum Formula

Given is the function  $x(t)$  whose spectrum is given by  $X(\omega) = 1/(1+|\omega|)^\alpha$  for some positive real number  $\alpha$ . The spectrum along with its replicas are shown for  $\alpha = 1.5$  in Figure P3.6-1. Determine a condition on  $\alpha$  such that  $\sum_{n \in \mathbb{Z}} x(n)$  converges.



**Figure P3.6-1:** (a) Continuous spectrum of the function  $x(t)$ . (b) Spectrum of discrete function  $x_n = x(n)$ . The replicas of the spectrum are present. In case of a slowly decaying function the discrete spectrum can become infinite.

## 3.7. Fourier transform of a Gaussian Function

Prove that (3.78) forms a Fourier transform pair.

(Hint: See the discussion following (3.78). You can use the fact that

$$\int_{t \in \mathbb{R}} e^{-t^2} dt = \sqrt{\pi}.$$

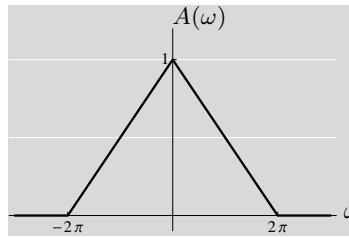
to specify the constant.)

## 3.8. Function Decay

Consider a signal  $a(t)$  having the Fourier transform  $A(\omega)$  given in the Fig. P3.8-1.

- (i) Write the expression for  $A(\omega)$ , and show that one possible solution with real values for the equation  $|\Phi(\omega)|^2 = A(\omega)$  is given by

$$\Phi(\omega) = \begin{cases} 0, & |\omega| > 2\pi \\ \sqrt{1 - \frac{|\omega|}{2\pi}}, & |\omega| \leq 2\pi \end{cases}.$$

**Figure P3.8-1:** Fourier transform  $A(\omega)$ .

- (ii) If
- $\varphi(t)$
- is the time domain version of
- $\Phi(\omega)$
- , show that

$$\langle \varphi(t), \varphi(t - n) \rangle = \delta_n.$$

(Hint: Make the analysis in the frequency domain; use Poisson's sum formula.)

- (iii) Consider the box function in frequency domain:

$$B(\omega) = \begin{cases} 0, & |\omega| > \pi \\ 1, & |\omega| \leq \pi \end{cases}.$$

Show that  $A(\omega) = \frac{1}{2\pi} B(\omega) * B(\omega)$  (convolution product). What can be said about the rate of decay of  $a(t)$ ?

**3.9. Decay of Fourier Transform and Smoothness**

Prove that if  $X(\omega)$  decays faster than  $1/|\omega|^{p+1}$  for large  $|\omega|$  as in (3.79b), then  $x(t) \in C^p$ . (Hint: Use (3.79a) and the differentiation property of the Fourier transform.)

**3.10. Lipschitz Regularity**

Assume that  $X(\omega)$  satisfies (3.82)

$$\int_{\omega \in \mathbb{R}} |X(\omega)| (1 + |\omega|^\alpha) d\omega < \infty.$$

- (i) Show that  $x(t)$  is bounded.  
 (ii) Show that  $x(t)$  is Lipschitz  $\alpha$ , that is, show (3.81):

$$\frac{|x(t) - x(t_0)|}{|t - t_0|^\alpha} \leq c,$$

for any  $t$  and  $t_0$  and  $0 \leq \alpha < 1$ .

(Hint: Express the above ratio in terms of the inverse Fourier transform, and divide the integral into two parts:  $|t - t_0|^{-1} \leq |\omega|$  and  $|t - t_0|^{-1} > |\omega|$ .)

- (iii) Show how to extend the above characterization for
- $r = n + \alpha$
- ,
- $n \in \mathbb{Z}^+$
- .

**3.11. Real Fourier Series**

Given is a  $2\pi$ -periodic, real-valued function  $x(t)$  with Fourier series coefficients  $X_k$ . Show that  $x(t)$  can be expressed as a real Fourier series

$$x(t) = a_0 + \sum_{k=1}^{\infty} (a_k \cos(kt) + b_k \sin(kt)),$$

where

$$\begin{aligned} a_0 &= \frac{1}{2\pi} \int_{-\pi}^{\pi} x(t) dt, \\ a_k &= \frac{1}{\pi} \int_{-\pi}^{\pi} x(t) \cos(kt) dt, \quad k \in \mathbb{Z}^+, \\ b_k &= \frac{1}{\pi} \int_{-\pi}^{\pi} x(t) \sin(kt) dt, \quad k \in \mathbb{Z}^+, \end{aligned}$$

and indicate the relationship between  $\{a_k, b_k\}$  and  $\{X_k\}$ .

### 3.12. Completeness of Fourier Series

As part of Theorem 3.15, prove (3.95) for continuous, real-valued  $\mathcal{L}^2$  functions on  $[-T/2, T/2]$ .

We do this in steps, introducing a deterministic autocorrelation function as a helper function. We start with  $x(t)$  on  $[-T/2, T/2]$ , assuming it is continuous and square integrable.

(i) Introduce  $x_T(t)$  as a periodized version of  $x(t)$  as in (3.72). Show that

$$a(t) = \int_{-T/2}^{T/2} x_T(t) x_T(t + \tau) dt,$$

is  $T$ -periodic and continuous.

(ii) Prove that (3.109) is indeed a Fourier-transform pair.

(iii) From the orthogonal projection theorem, we can write

$$\int_{-T/2}^{T/2} |x(t)|^2 dt = \sum_{k=-N}^N |X_k|^2 + \int_{-T/2}^{T/2} |x(t) - \hat{x}_N(t)|^2 dt, \quad (\text{P3.12-1})$$

with  $\hat{x}_N(t)$  as in (3.94). It suffices to show that, as  $N \rightarrow \infty$ ,

$$\int_{-T/2}^{T/2} |x(t)|^2 dt = \sum_{k \in \mathbb{Z}} |X_k|^2 \quad (\text{P3.12-2})$$

to prove (3.95) and therefore completeness.

From (P3.12-1), verify that  $\sum_{k \in \mathbb{Z}} |X_k|^2 < \infty$ , and since  $a(t)$  is continuous, we can use Theorem 3.17 to prove that

$$a(t) = \sum_{k=-\infty}^{\infty} |X_k|^2 e^{j(2\pi/T)kt},$$

for all  $t$ .

(iv) For the particular value  $t = 0$ , show (3.95).

### 3.13. Uniqueness of Fourier Series

Consider  $T$ -periodic functions  $x_1(t)$  and  $x_2(t)$ , each with an absolutely-integrable period.

(i) Show that if  $X_{1,k} = X_{2,k}$  for all  $k$ , then they are equal almost everywhere.

(ii) Show that if they are continuous, then they are equal everywhere.

### 3.14. Integration of Fourier Series

Given is a  $2\pi$ -periodic, real-valued, zero-mean function  $x(t)$  with real Fourier series as in Exercise 3.11. Consider the primitive of  $x(t)$ , with period

$$X(t) = \int_0^t x(\tau) d\tau, \quad t \in [-\pi, \pi].$$

Show that the real Fourier series of  $X(t)$  is

$$X(t) = A_0 + \sum_{k=1}^{\infty} \left( \frac{a_k}{k} \sin(kt) - \frac{b_k}{k} \cos(kt) \right),$$

where  $A_0 = \sum_{k=1}^{\infty} b_k/k$  is finite, and  $\{a_k, b_k\}$  are the real Fourier series coefficients of  $x(t)$ .

### 3.15. Convolution Rule for Fourier Series

Given is a  $T$ -periodic signal  $x(t)$  and a stable filter  $h(t)$ . Prove that the following is a Fourier series pair:

$$y(t) = \int_{\tau \in \mathbb{R}} h(\tau) x(t - \tau) dt \xleftrightarrow{\text{FS}} Y_k = H\left(\frac{2\pi}{T}k\right) X_k,$$

where  $y(t)$  is the  $T$ -periodic output of the convolution.

3.16. *Fourier Series of a Triangle Wave*

In (3.115), the Fourier series of the triangle wave was computed using the integration property. Obtain the same result using the following alternatives:

- (i) Direct computation using the definition of the Fourier series (3.90a).
- (ii) Using the Fourier transform of one period of the triangle (3.49f) and sampling (3.98).
- (iii) Using the convolution property.  
(*Hint:* Use a square wave and a filter that is the indicator function of  $[0, 1/2]$ .)

3.17. *Sawtooth Function and Gibbs Phenomenon*

Given is the sawtooth function of period  $T = 1$ , with one period given by

$$x(t) = \begin{cases} 1/2 - t, & -1/2 < t < 0; \\ 0, & t = 0; \\ -1/2 - t, & 0 < t < 1/2. \end{cases} \quad (\text{P3.17-1})$$

Compute the Fourier series coefficients of  $x(t)$  and comment on possible Gibbs phenomenon.



## Chapter 4

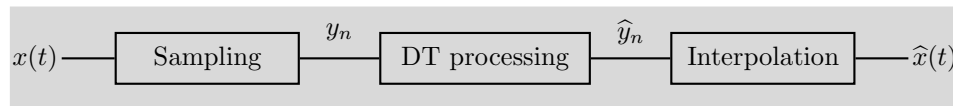
# Sampling and Interpolation

## Contents

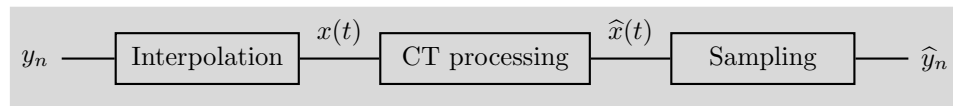
4.1	Introduction . . . . .	378
4.2	Finite-Dimensional Vectors . . . . .	385
4.3	Sequences . . . . .	393
4.4	Functions . . . . .	404
4.5	Periodic Functions . . . . .	426
4.6	Stochastic Vectors and Processes . . . . .	436
4.7	Computational Aspects . . . . .	440
	Chapter at a Glance . . . . .	445
	Historical Remarks . . . . .	447
	Further Reading . . . . .	447
	Exercises with Solutions . . . . .	448
	Exercises . . . . .	452

The previous two chapters dealt with discrete-time sequences indexed by integers and continuous-time functions of a real variable. The primary purpose of the present chapter is to link these two worlds. This is done through *sampling*, which produces a sequence from a function, and *interpolation*, which produces a function from a sequence. The ability to sample a function, manipulate the resulting sequence with a discrete-time system, and then interpolate to produce a function, is the foundation of digital signal processing. Conversely, the ability to interpolate a sequence to create a function, manipulate the resulting function with a continuous-time system, and then sample to produce a sequence, is the foundation of digital communications. These interactions conceptually position the chapter as a bridge between Chapters 2 and 3 as illustrated in Figure 4.1.

Given a continuous-time function, one can associate a sequence by simply taking samples (evaluating or measuring the function) uniformly in time. Classical sampling theory places a bandwidth restriction on the function so that the samples are a faithful representation of the function leading to the concept of Nyquist rate,



(a) Digital signal processing.



(b) Digital communications.

**Figure 4.1:** Sampling and interpolation in signal processing and communications. (a) Digital signal processing: sampling produces a discrete-time sequence from a continuous-time function, which is then processed with a discrete-time system and interpolated. (b) Digital communications: interpolation produces a continuous-time function from a discrete-time sequence, which is then processed with a continuous-time system and then sampled.

a minimum rate of sampling so that changes in the function are captured. We will develop this result in detail, but will also see it as a special case of a more general theory involving shift-invariant subspaces. Our approach is intimately tied to basis expansions and subspaces: sampling followed by interpolation projects to a subspace, while interpolation alone embeds information within a subspace in a higher-dimensional space.

## 4.1 Introduction

In Chapters 2 and 3, we saw a pair of bijections between sequences and functions:

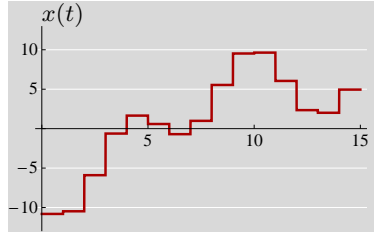
$$\begin{array}{lll} \text{discrete-time sequence} & \xleftrightarrow{\text{DTFT}} & \text{periodic Fourier-domain function,} \\ \text{periodic continuous-time function} & \xleftrightarrow{\text{FS}} & \text{discrete Fourier-domain sequence.} \end{array}$$

The first associates a periodic function, the discrete-time Fourier transform  $X(e^{j\omega})$ , to a discrete-time sequence  $x_n$ , and the second associates a sequence, the Fourier series  $X_k$ , to a periodic continuous-time function  $x(t)$ .

While these are important connections, they are different in spirit from sampling and interpolation, which both operate within the time domain. Even when sampling and interpolation do not only operate between discrete and continuous domains, that instance is the most common one:

$$\begin{array}{ccc} \text{discrete-time sequence} & \begin{array}{c} \xleftrightarrow{\text{interpolation}} \\ \xleftarrow{\text{sampling}} \end{array} & \text{continuous-time function.} \end{array}$$

In this section, we capture the main themes of the chapter by running through one representative example that includes going from continuous time to discrete time and back. This expands upon Examples 1.14(i) and 1.15(iii).



**Figure 4.2:** Piecewise-constant functions over unit intervals.

**Subspace of Functions** Consider the space  $S$  of functions that are piecewise constant over unit intervals as in Figure 4.2,

$$x(t) = x(n) \quad \text{for all } t \in [n, n+1), \quad n \in \mathbb{Z}. \quad (4.1)$$

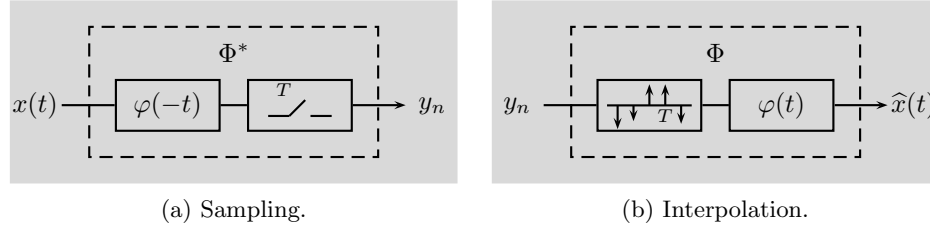
The space  $S$  is shift-invariant with respect to integer shifts, that is, for  $x(t) \in S$  and any  $k \in \mathbb{Z}$ ,  $x(t - k)$  also belongs to  $S$ . Because of (4.1), functions in  $S$  are in one-to-one correspondence with sequences. If  $\varphi(t) = \chi_{[0,1)}(t)$  is the indicator function of the unit interval, the set  $\{\varphi(t - k)\}_{k \in \mathbb{Z}}$  is an orthonormal basis for  $S$ .

Instead of unit-length interval, we could consider any interval of length  $T > 0$ . This time, the set  $\{(1/\sqrt{T})\varphi(t/T - k)\}_{k \in \mathbb{Z}}$  is an orthonormal basis for  $S$ , the space of piecewise-constant functions over intervals  $[kT, (k+1)T)$ . The interval length  $T$  is called the *sampling period* and  $1/T$  is called the *sampling rate*. Changing the sampling period  $T$  allows us to adjust the sampling rate to the function at hand (sample more often for a fast-varying function or sample less often for a slowly-varying function).

**Sampling** Suppose we want to measure a continuous-time function  $x(t)$  to obtain a sequence of real numbers  $y_n$  describing  $x(t)$ . While the device we employ might be able to take a measurement of  $x(t)$  at a single point in time, this measurement would be sensitive to noise. Instead, it is more robust to measure an integral; for all  $n \in \mathbb{Z}$ ,

$$\begin{aligned} y_n &= \int_n^{n+1} x(t) dt \stackrel{(a)}{=} \int_{-\infty}^{\infty} x(t) \varphi(t - n) dt \\ &= \varphi(-t) * x(t)|_{t=n} = \langle \varphi(t - n), x(t) \rangle_t \stackrel{(b)}{=} (\Phi^* x)_n, \end{aligned} \quad (4.2)$$

where (a) follows from the definition of the indicator function  $\chi_{[0,1)}(t) = \varphi(t)$ ; and in (b)  $\Phi^*$  denotes the sampling operator illustrated in Figure 4.3(a) with  $T = 1$ . In other words, this sampling operator is implemented using filtering by  $\varphi(-t)$  and sampling at integer instants. Since square integrability of  $x(t)$  implies square summability of  $y_n$ ,  $\Phi^*$  is a mapping from  $\mathcal{L}^2(\mathbb{R})$  to  $\ell^2(\mathbb{Z})$  (see Example 1.14(i)).



**Figure 4.3:** Sampling and interpolation for piecewise-constant functions. (a) Sampling is filtering by  $\varphi(-t)$  followed by sampling at  $t = n$ . (b) Interpolation is pointwise multiplication by the Dirac delta comb function  $s_1(t)$  followed by filtering by  $\varphi(t)$ .

**Interpolation** Interpolation generates a function  $\hat{x}(t)$  from a sequence  $y_n$ . One way to do this is to form

$$\hat{x}(t) = \sum_{n \in \mathbb{Z}} y_n \varphi(t - n) = (\Phi y)(t), \quad (4.3)$$

where  $\Phi$  denotes the interpolation operator illustrated in Figure 4.3(b) with  $T = 1$ . In other words, the sequence  $y_n$  is multiplied pointwise by the Dirac delta comb function  $s_1(t)$  from (3.7), before being filtered by  $\varphi(t)$ . From Examples 1.14(i) and 1.15(iii), we know that this operator is the adjoint of the sampling one; thus our choice to call it  $\Phi$ . Since  $\hat{x}(t)$  is constant on intervals  $[n, n + 1)$ ,  $n \in \mathbb{Z}$ , a calculation similar to Example 1.14(i) shows that square summability of  $y_n$  implies square integrability of  $\hat{x}(t)$ ; thus,  $\Phi$  is a mapping from  $\ell^2(\mathbb{Z})$  to  $\mathcal{L}^2(\mathbb{R})$ .

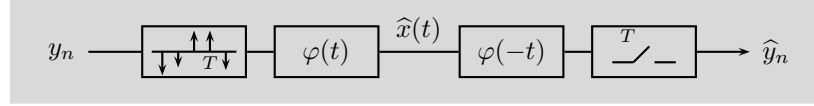
**Interpolation Followed by Sampling: Sequence Recovery** Figure 4.4(a) with  $T = 1$  depicts interpolation followed by sampling. Because of the specific choice of sampling and interpolation operators, we have that  $\Phi^* \Phi = I$ ; in other words, any sequence  $y_n \in \ell^2(\mathbb{Z})$  is recovered perfectly when the interpolated function computed through (4.3) is used in the sampling formula (4.2),

$$\Phi^* \Phi y_n = y_n, \quad y_n \in \ell^2(\mathbb{Z}).$$

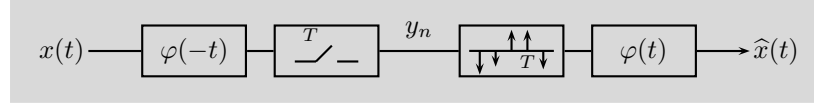
Another choice for sampling or interpolation would not have necessarily lead to perfect recovery. Suppose that the interpolation operator had been different, for example, linear interpolation instead of constant one, that is  $\varphi(t)$  had been  $|t|$  on the unit interval and 0 otherwise; then,  $\Phi^* \Phi \neq I$ . We will discuss such instances later in the chapter.

**Sampling Followed by Interpolation: Function Recovery** Figure 4.4(b) with  $T = 1$  depicts sampling followed by interpolation. Because of the specific choice of sampling and interpolation operators as well as the fact that the function  $x(t)$  belongs to  $S$ , the function is recovered perfectly when the samples computed through (4.2) are used in the interpolation formula (4.3),

$$\Phi \Phi^* x(t) = x(t), \quad x(t) \in S \subset \mathcal{L}^2(\mathbb{R}).$$

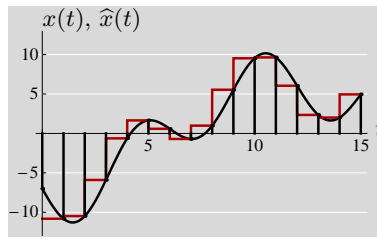


(a) Interpolation followed by sampling.

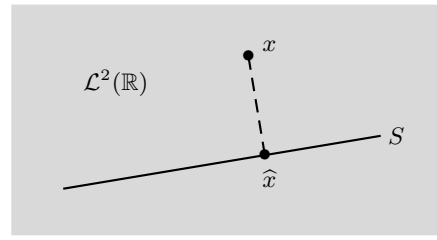


(b) Sampling followed by interpolation.

**Figure 4.4:** Sampling and interpolation for piecewise-constant functions. (a) Interpolation as in (4.3) followed by sampling as in (4.2),  $\Phi^* \Phi$ , recovers the input sequence perfectly for any  $y_n \in \ell^2(\mathbb{Z})$ , that is,  $\hat{y}_n = y_n$ . (b) Sampling as in (4.2) followed by interpolation as in (4.3),  $\Phi \Phi^*$ , recovers the input function perfectly when  $x(t) \in S$ , that is,  $\hat{x}(t) = x(t)$ .



(a)



(b)

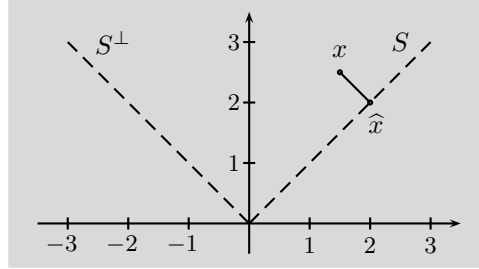
**Figure 4.5:** Least-squares approximation of an arbitrary function  $x(t)$  by a piecewise-constant approximation  $\hat{x}(t) \in S$ . (a) Example function and its piecewise-constant approximation. (b) Conceptual depiction of orthogonal projection.

Unlike for interpolation followed by sampling, sampling followed by interpolation poses restrictions on the input function to guarantee perfect recovery; here, the input function must belong to the subspace  $S \in \mathcal{L}^2(\mathbb{R})$ . Both the sequence recovery property and the function recovery property depend on having a proper match between sampling and interpolation operators as we will discuss later in the chapter.

When  $x(t) \notin S$ , sampling followed by interpolation,  $P = \Phi \Phi^*$ , does not act as the identity on  $x(t)$ , and we only obtain an approximation. Finding the closest function in  $S$  (where distance is measured with the  $\mathcal{L}^2$  norm) is simple because of Hilbert-space geometry. We find that

$$\begin{aligned} P^2 &= \Phi \Phi^* \Phi \Phi^* \stackrel{(a)}{=} \Phi \Phi^* = P, \\ P^* &= (\Phi \Phi^*)^* = \Phi \Phi^* = P, \end{aligned}$$

where (a) follows from  $\Phi^* \Phi = I$ . In other words,  $P$  is idempotent and self-adjoint, that is,  $P$  is an orthogonal projection operator. The projection theorem, Theorem 1.26, then states that given an arbitrary  $x(t) \in \mathcal{L}^2(\mathbb{R})$ , sampling followed by



**Figure 4.6:** Best least-squares approximation of  $x = (1/2) [3 \ 5]^T$  via the orthogonal projection operator (4.5) in  $\mathbb{R}^2$ .

interpolation results in  $\hat{x}(t) = \Phi\Phi^*x(t)$ , the best least-squares approximation of  $x(t)$  in  $S$  (see Figure 4.5).

Another way to verify the best least-squares approximation property is to note that the set  $\{\varphi(t - k)\}_{k \in \mathbb{Z}}$  is an orthonormal basis for  $S$  and that (4.3) is an orthonormal basis expansion formula. Thus, the approximation property follows from Theorem 1.40. Finally, one can explicitly verify that computing the average of a function over an interval minimizes the  $\mathcal{L}^2$  norm of the error of a piecewise-constant approximation to the function (see Exercise 4.1).

When the interval length  $T \neq 1$ , and for any fixed  $T > 0$ , there are functions  $x(t) \in \mathcal{L}^2(\mathbb{R})$  that differ appreciably from the closest function  $\hat{x}_T(t)$  that is piecewise constant over intervals  $[kT, (k+1)T)$ . However, considering all  $T > 0$ , these piecewise-constant functions are dense in  $\mathcal{L}^2(\mathbb{R})$ , and the approximation error between  $\hat{x}_T(t)$  and  $x(t)$  goes to zero as  $T \rightarrow 0$ . The rate at which this error goes to zero is an important parameter; it indicates the *approximation power* of the sampling and interpolation scheme (see Solved Exercise 4.1).

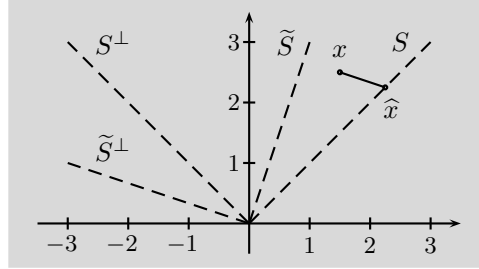
**Sampling and Interpolation with Nonorthogonal Vectors** In the discussion so far, we have seen that the sampling and interpolation operators are adjoints of each other, making  $P = \Phi\Phi^*$  an orthogonal projection operator. This does not have to be the case, and throughout this chapter we will see instances of general sampling and interpolation operators,  $\tilde{\Phi}^*$  and  $\Phi$ , respectively. To get a feel for these, we will look into the simplest setting, vectors in  $\mathbb{R}^2$ . To be able to compare orthogonal to nonorthogonal case, we start first with orthogonal and then follow with nonorthogonal one.

- (i) Consider first the sampling and interpolation operators to be:

$$\Phi^* = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \end{bmatrix}, \quad \Phi = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \quad (4.4a)$$

The null space of the sampling operator and its orthogonal complement, which is the same as the range of the interpolation operator, are

$$S^\perp = \mathcal{N}(\Phi^*) = \left\{ \alpha \begin{bmatrix} -1 \\ 1 \end{bmatrix} \right\}, \quad S = \mathcal{R}(\Phi) = \left\{ \alpha \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\}, \quad \alpha \in \mathbb{R}. \quad (4.4b)$$



**Figure 4.7:** Approximation of  $x = (1/2) [3 \ 5]^T$  via the projection operator (4.7) in  $\mathbb{R}^2$ .

Clearly, interpolation followed by sampling leads to identity,

$$\Phi^* \Phi = 1.$$

What is more interesting is that sampling followed by interpolation,  $P = \Phi \Phi^*$ , is an orthogonal projection operator,

$$P = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \quad P^2 = P, \quad P^* = P. \quad (4.5)$$

Then, for any given  $x \in \mathbb{R}^2$ ,  $\hat{x} = Px$  is the best least-squares approximation of  $x$  onto  $S$ ; when  $x \in S$ , then  $\hat{x} = x$ . Figure 4.6 illustrates these spaces as well as the approximation for  $x = (1/2) [3 \ 5]^T$ .

- (ii) We keep the interpolation operator from (4.4a) the same and choose the sampling operator to be

$$\tilde{\Phi}^* = \frac{1}{2\sqrt{2}} [1 \ 3]. \quad (4.6a)$$

The null space of the sampling operator and its orthogonal complement (not anymore the same as the range of the interpolation operator), are

$$\tilde{S}^\perp = \mathcal{N}(\tilde{\Phi}^*) = \left\{ \alpha \begin{bmatrix} -3 \\ 1 \end{bmatrix} \right\}, \quad \tilde{S} = \left\{ \alpha \begin{bmatrix} 1 \\ 3 \end{bmatrix} \right\}, \quad \alpha \in \mathbb{R}. \quad (4.6b)$$

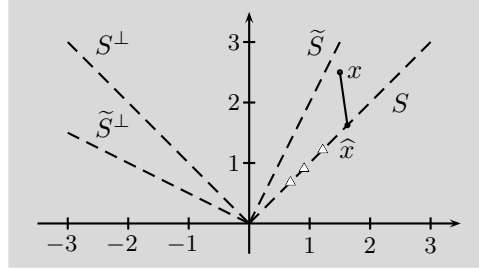
Again, interpolation followed by sampling leads to identity,

$$\Phi^* \Phi = 1.$$

However, sampling followed by interpolation,  $P = \Phi \Phi^*$ , is not an orthogonal projection operator anymore, albeit it is still a projection operator,

$$P = \frac{1}{4} \begin{bmatrix} 1 & 3 \\ 1 & 3 \end{bmatrix}, \quad P^2 = P, \quad P^* \neq P. \quad (4.7)$$

When  $x \in S$ , then  $\hat{x} = x$ . Figure 4.7 illustrates these spaces as well as the approximation for  $x = (1/2) [3 \ 5]^T$ .



**Figure 4.8:** Operator (4.9) that is not a projection applied to  $x = (1/2) [3 \ 5]^T$ . Triangles denote  $P^2x$ ,  $P^3x$ ,  $P^4x$ .

- (iii) We still keep the interpolation operator from (4.4a) the same and choose the sampling operator to be

$$\tilde{\Phi}^* = \frac{1}{2\sqrt{2}} \begin{bmatrix} 1 & 2 \end{bmatrix}. \quad (4.8a)$$

The null space of the sampling operator and its orthogonal complement are

$$\tilde{S}^\perp = \mathcal{N}(\tilde{\Phi}^*) = \left\{ \alpha \begin{bmatrix} -2 \\ 1 \end{bmatrix} \right\}, \quad \tilde{S} = \left\{ \alpha \begin{bmatrix} 1 \\ 2 \end{bmatrix} \right\}, \quad \alpha \in \mathbb{R}. \quad (4.8b)$$

This time, interpolation followed by sampling does not lead to identity,

$$\Phi^* \Phi \neq 1.$$

Sampling followed by interpolation,  $P = \Phi \Phi^*$ , is not even a projection operator anymore,

$$P = \frac{1}{4} \begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix}, \quad P^2 = \frac{3}{16} \begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix} \neq P, \quad P^* \neq P. \quad (4.9)$$

Since this is not a projection operator, it goes along  $S$  away from  $Px$  if applied again. In general,

$$P^k = \frac{1}{4} \left( \frac{3}{4} \right)^{k-1} \begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix},$$

meaning that eventually, the point will be mapped to 0. Figure 4.8 illustrates this effect<sup>77</sup> as well as the spaces involved for  $x = (1/2) [3 \ 5]^T$ .

### Chapter Outline

The chapter follows this brief introduction for different spaces: Section 4.2 develops sampling theory from the perspective of linear operators (matrices) in finite-dimensional spaces; we do that for orthonormal sets of vectors first, followed by

<sup>77</sup>Here, the points move along  $S$  closer to the origin because the operator has eigenvalues smaller than 1 in absolute value. Choosing a different scaling in (4.8a), for example,  $1/2$  instead of  $1/(2\sqrt{2})$ , would make the points along  $S$  to infinity. Note that a scaling of  $1/\sqrt{5}$  makes it idempotent.



nonorthogonal ones. Sections 4.3 and 4.4 do the same for sequences and functions, respectively. In these settings, the use of LSI filtering naturally leads to the sampling theory for shift-invariant subspaces, as well as a special case of bandlimited sequences and functions. The celebrated sampling theorem for bandlimited functions is presented both with a classical justification and as an orthonormal expansion of the subspace of bandlimited functions of a given bandwidth. We also consider multichannel sampling. Section 4.6 considers the stochastic setting, and finally, Section 4.7 concludes with the discussion of computational aspects.

*Notation used in this chapter:* Starting with Section 4.3, we use  $\varphi$  and  $g$  in parallel to denote sequences/functions from the points of view of expansions in bases/projections onto subspaces as well as signal processing using filtering. We do that so that we show connections between these points of view.  $\square$

## 4.2 Finite-Dimensional Vectors

We now look into the interpretation of sampling and interpolation when the larger space is the space of finite-dimensional vectors,  $\mathbb{C}^M$ . Subspaces of this larger space are finite-dimensional spaces  $\mathbb{C}^N$ , with  $N < M$  (and often  $N \ll M$ ). Then sampling will take  $M$  values and produce  $N < M$  values, while interpolation will take  $N$  values and produce  $M > N$  values.

### 4.2.1 Sampling and Interpolation with Orthonormal Vectors

**Sampling** Sampling is a linear operator from  $\mathbb{C}^M$  to  $\mathbb{C}^N$ , so it can be represented by an  $N \times M$  matrix  $\Phi^*$ . For a given input vector  $x \in \mathbb{C}^M$ , the sampling output is a vector  $y \in \mathbb{C}^N$ ,

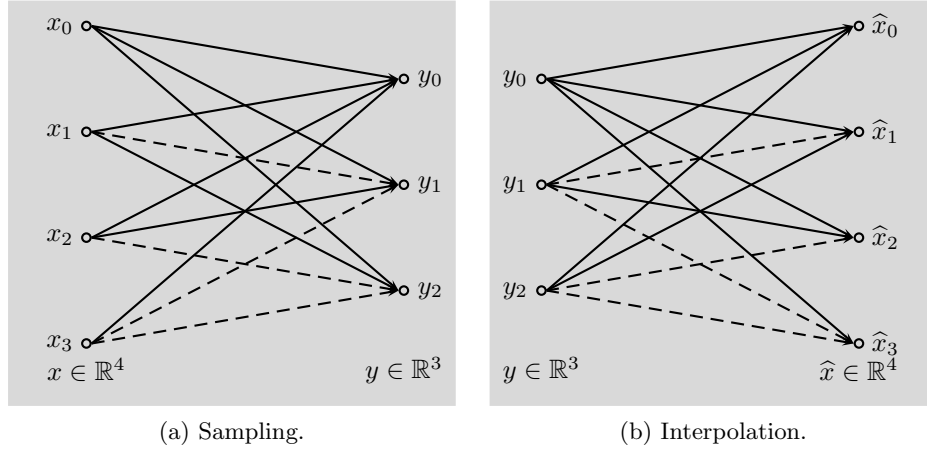
$$y = \begin{bmatrix} \langle x, \varphi_0 \rangle \\ \langle x, \varphi_1 \rangle \\ \vdots \\ \langle x, \varphi_{N-1} \rangle \end{bmatrix}_{N \times 1} = \begin{bmatrix} \varphi_0^* \\ \varphi_1^* \\ \vdots \\ \varphi_{N-1}^* \end{bmatrix}_{N \times M} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{M-1} \end{bmatrix}_{M \times 1} = \Phi^* x. \quad (4.10)$$

In the above matrix,  $\varphi_k^*$  is the  $k$ th row of  $\Phi^*$ ; we assume  $\varphi_k^*$  to be orthonormal,

$$\langle \varphi_n, \varphi_k \rangle = \delta_{n-k} \quad \Leftrightarrow \quad \Phi^* \Phi = I, \quad (4.11)$$

and thus,  $\Phi^*$  has maximum rank  $N$ . Then, the sampling operator has an  $(M - N)$ -dimensional null space,  $\mathcal{N}(\Phi^*)$ ; the set  $\{\varphi_k\}_{k=0}^{N-1}$  spans its orthogonal complement,  $S = \mathcal{N}(\Phi^*)^\perp = \text{span}(\{\varphi_k\}_{k=0}^{N-1})$ . In other words, when a vector  $x \in \mathbb{C}^M$  is sampled, the component that remains is in  $S$  and is captured by  $\Phi^* x$ ; the component that is lost due to sampling is in the null space  $S^\perp$ . We illustrate this with an example.

**EXAMPLE 4.1 (SAMPLING IN  $\mathbb{R}^4$ )** Let us define sampling of  $x \in \mathbb{R}^4$  to obtain three samples  $y \in \mathbb{R}^3$  as in Figure 4.9(a), where solid lines have weight  $1/2$ , while



**Figure 4.9:** Sampling and interpolation in  $\mathbb{R}^4$  with orthonormal vectors. Solid lines have weight  $1/2$ , while dashed lines have weight  $-1/2$ .

dashed lines have weight  $-1/2$ ; for example,  $y_1 = (x_0 - x_1 + x_2 - x_3)/2$ . Then, the sampling operator can be written as

$$\Phi^* = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \end{bmatrix}. \quad (4.12a)$$

With  $\alpha \in \mathbb{R}$ ,  $\alpha_k \in \mathbb{R}$ , the null space of  $\Phi^*$  and its orthogonal complement are

$$\mathcal{N}(\Phi^*) = \left\{ \alpha \begin{bmatrix} 1 \\ -1 \\ -1 \\ 1 \end{bmatrix} \right\}, \quad S = \left\{ \alpha_0 \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \alpha_1 \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix} + \alpha_2 \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix} \right\}. \quad (4.12b)$$

Applying the sampling operator to an arbitrary vector in  $\mathbb{R}^4$ ,

$$\Phi^* \begin{bmatrix} 2 \\ 0 \\ 0 \\ 2 \end{bmatrix} = \Phi^* \left( \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{\in S} + \underbrace{\begin{bmatrix} 1 \\ -1 \\ -1 \\ 1 \end{bmatrix}}_{\in S^\perp} \right) = \Phi^* \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \underbrace{\Phi^* \begin{bmatrix} 1 \\ -1 \\ -1 \\ 1 \end{bmatrix}}_{=0} = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Thus, one component of the vector, the one belonging to the null space  $S^\perp$ , was mapped into 0 and lost.

**Interpolation** Interpolation is a linear operator from  $\mathbb{C}^N$  to  $\mathbb{C}^M$ ,  $N < M$ , so it can be represented by an  $M \times N$  matrix; we choose that matrix to be the adjoint

of the sampling operator,  $\Phi$ , as we have done in the introductory example. For a given input vector  $y \in \mathbb{C}^N$ , the interpolation output is a vector  $\hat{x} \in \mathbb{C}^M$ ,

$$\hat{x} = [\varphi_0 \ \varphi_1 \ \dots \ \varphi_{N-1}]_{M \times N} \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{N-1} \end{bmatrix}_{N \times 1} = \Phi y = \sum_{k=0}^{N-1} y_k \varphi_k. \quad (4.13)$$

In the above matrix,  $\varphi_k$  is the  $k$ th column of  $\Phi$ . As was true for  $\Phi^*$ ,  $\Phi$  has maximum rank  $N$ . Thus, the interpolation operator has an  $N$ -dimensional range  $S$ , which is a proper subspace of  $\mathbb{C}^M$  and is given by  $S = \text{span}(\{\varphi_k\}_{k=0}^{N-1})$ . This subspace is, of course, the same as the orthogonal complement of the null space of the sampling operator, as we have seen earlier.

**EXAMPLE 4.2 (INTERPOLATION IN  $\mathbb{R}^4$ )** From (4.12a), the interpolation operator is

$$\Phi = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \\ 1 & -1 & -1 \end{bmatrix}, \quad (4.14)$$

all illustrated in Figure 4.9(b). The range of this operator is  $S$  (the same as the orthogonal complement of the null space of the sampling operator in (4.12b)).

**Interpolation Followed by Sampling** Interpolation followed by sampling is described by  $\Phi^* \Phi$ , which maps from the smaller space,  $\mathbb{C}^N$ , to itself. Since by assumption (4.11) holds,  $y$  is perfectly recovered. Equation (4.11) also shows that the condition for perfect recovery is the same as the set of vectors  $\{\varphi_k\}_{k=0}^{N-1}$  being orthonormal, as in (1.83). This set of vectors is not a basis for  $\mathbb{C}^M$  (too few vectors); instead, it is an orthonormal basis for the  $N$ -dimensional subspace  $S$  it spans.

**Sampling Followed by Interpolation** Sampling followed by interpolation is described by  $P = \Phi \Phi^*$ . Intuitively, this is a more difficult sequence of operations to recover from perfectly, as sampling, unless the input is in  $S$ , leads to a loss.

Given our choice of sampling and interpolation operators,

$$\begin{aligned} P^2 &= \Phi \Phi^* \Phi \Phi^* \stackrel{(a)}{=} \Phi \Phi^* = P, \\ P^* &= (\Phi \Phi^*)^* = \Phi \Phi^* = P, \end{aligned}$$

where (a) follows from (4.11). In other words,  $P$  is idempotent and self-adjoint, that is,  $P$  is an orthogonal projection operator. Then, by Theorem 1.26,  $Px$  is the best least-squares approximation of  $x$  in  $S$ ; if  $x \in S$ , sampling followed by interpolation will perfectly recover  $x$ :

**THEOREM 4.1 (RECOVERY FOR FINITE-DIMENSIONAL VECTORS)** Given is sampling followed by interpolation with sampling operator  $\Phi^*$  from (4.24) and interpolation operator  $\Phi$  satisfying (4.25). Then, with  $S = \mathcal{R}(\Phi)$ ,

$$\hat{x} = \Phi\Phi^*x \quad (4.15)$$

is the best least-squares approximation of  $x$  in  $S$ , that is,

$$\hat{x} = \min_{x_S \in S} \|x - x_S\|^2, \quad \hat{x} - x \perp S.$$

When  $x \in S$ , then  $\hat{x} = x$ .

**EXAMPLE 4.3 (SAMPLING FOLLOWED BY INTERPOLATION IN  $\mathbb{R}^4$ )** We now illustrate the above result.

Choose first  $x \in S$ . Then the output is

$$\begin{aligned} \hat{x} &= Px = \Phi\Phi^*x \stackrel{(a)}{=} \Phi\Phi^*(\alpha_0\varphi_0 + \alpha_1\varphi_1 + \alpha_2\varphi_2) \\ &\stackrel{(b)}{=} \frac{1}{4} \begin{bmatrix} 3 & 1 & 1 & -1 \\ 1 & 3 & -1 & 1 \\ 1 & -1 & 3 & 1 \\ -1 & 1 & 1 & 3 \end{bmatrix} \frac{1}{2} \left( \alpha_0 \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \alpha_1 \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix} + \alpha_2 \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix} \right) \\ &= \frac{1}{2} \left( \alpha_0 \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \alpha_1 \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix} + \alpha_2 \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix} \right) = x, \end{aligned}$$

where (a) follows because  $x \in S$  and can thus be expressed as a linear combination of its basis vectors,  $\{\varphi_k\}_{k=0}^2$ ; and (b) from the expression for  $\Phi$ , (4.14).

Choose next  $x \notin S$ , vector  $x = [2 \ 0 \ 0 \ 2]^T$  from Example 4.1. The best we can do now is compute an approximation. Applying  $P = \Phi\Phi^*$ ,

$$\hat{x} = Px = \Phi\Phi^*x = \frac{1}{4} \begin{bmatrix} 3 & 1 & 1 & -1 \\ 1 & 3 & -1 & 1 \\ 1 & -1 & 3 & 1 \\ -1 & 1 & 1 & 3 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ 0 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix},$$

which clearly belongs to  $S$  since it equals the first basis vector  $\varphi_0$ . It is also the closest vector in  $S$ , since the error between  $x$  and  $\hat{x}$  is

$$x - \hat{x} = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 2 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ -1 \\ 1 \end{bmatrix} \perp S.$$

Finite-dimensional vectors can be seen as finite-length sequences from Chapter 2. In that case, we know that the DFT will be an appropriate Fourier transform

when these sequences are circularly extended (or, viewed as periodic sequences). We could then define bandlimited subspaces of  $\mathbb{C}^M$  (similarly what we will do in the following sections for sequences and functions). Exercise 4.2 explores sampling and interpolation in such subspaces.

### 4.2.2 Sampling and Interpolation with Nonorthogonal Vectors

What we have seen thus far is a rather classical take on sampling and interpolation; we now expand it a bit to include nonorthogonal vectors. As in our discussion of orthonormal and biorthogonal pairs of bases, nonorthogonal vectors make the geometry more complicated; the sampling and interpolation operators are no longer adjoints of each other, and the appropriate spaces we discussed earlier, the range of the interpolation operator as well as the orthogonal complement of the null space of the sampling operator, are no longer the same.

**Sampling** Again, sampling is represented by an  $N \times M$  matrix as in (4.10), but this time containing rows that are not orthogonal. We call these rows  $\tilde{\varphi}_k^*$  and the corresponding sampling matrix  $\tilde{\Phi}^*$ . Thus, for a given input vector  $x \in \mathbb{C}^M$ , the sampling output is a vector  $y \in \mathbb{C}^N$ ,

$$y = \begin{bmatrix} \langle x, \tilde{\varphi}_0 \rangle \\ \langle x, \tilde{\varphi}_1 \rangle \\ \vdots \\ \langle x, \tilde{\varphi}_{N-1} \rangle \end{bmatrix}_{N \times 1} = \begin{bmatrix} \tilde{\varphi}_0^* \\ \tilde{\varphi}_1^* \\ \vdots \\ \tilde{\varphi}_{N-1}^* \end{bmatrix}_{N \times M} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{M-1} \end{bmatrix}_{M \times 1} = \tilde{\Phi}^* x. \quad (4.16)$$

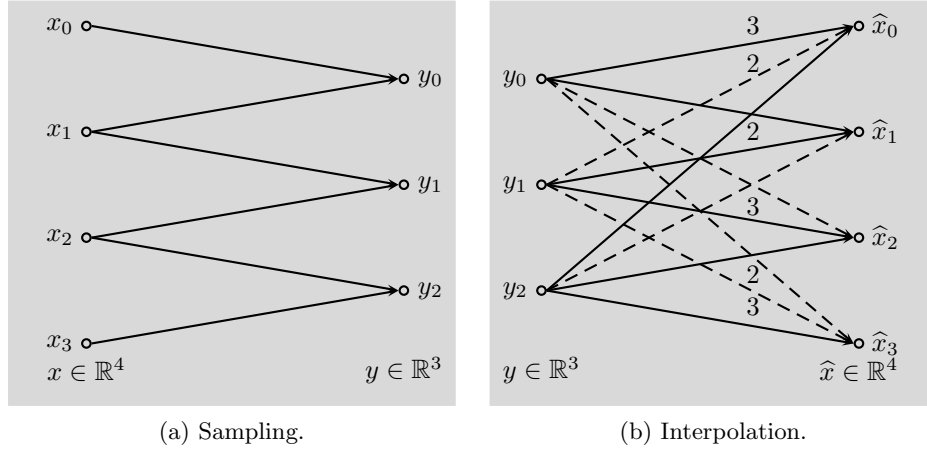
We again assume  $\tilde{\Phi}^*$  to have maximum rank  $N$ . Thus, the sampling operator has an  $(M - N)$ -dimensional null space,  $\mathcal{N}(\tilde{\Phi}^*)$ ; the set  $\{\tilde{\varphi}_k\}_{k=0}^{N-1}$  spans its orthogonal complement,  $\tilde{S} = \mathcal{N}(\tilde{\Phi}^*)^\perp = \text{span}(\{\tilde{\varphi}_k\}_{k=0}^{N-1})$ . In other words, when a vector  $x \in \mathbb{C}^M$  is sampled, the component that remains is in  $\tilde{S}$  and is captured by  $\tilde{\Phi}^* x$ ; the component that is lost due to sampling is in the null space  $\tilde{S}^\perp$ .

**EXAMPLE 4.4 (SAMPLING IN  $\mathbb{R}^4$ )** Let us define sampling of  $x \in \mathbb{R}^4$  to obtain three samples  $y \in \mathbb{R}^3$  as using the midpoints of neighboring pairs of samples, as shown in Figure 4.10(a). Sample  $y_k$  is the average of  $x_k$  and  $x_{k+1}$ , so the sampling operator can be written as

$$\tilde{\Phi}^* = \frac{1}{2} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}. \quad (4.17a)$$

With  $\beta \in \mathbb{R}$ ,  $\beta_k \in \mathbb{R}$ , the null space of  $\tilde{\Phi}^*$  and its orthogonal complement are

$$\mathcal{N}(\tilde{\Phi}^*) = \left\{ \beta \begin{bmatrix} -1 \\ 1 \\ -1 \\ 1 \end{bmatrix} \right\}, \quad \tilde{S} = \frac{1}{2} \left\{ \beta_0 \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} + \beta_1 \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} + \beta_2 \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \right\}. \quad (4.17b)$$



**Figure 4.10:** Sampling and interpolation in  $\mathbb{R}^4$  with nonorthogonal vectors. Solid lines have weight  $1/2$ , while dashed lines have weight  $-1/2$ ; factors above the line multiply  $1/2$ .

**Interpolation** Again, interpolation is represented by an  $M \times N$  matrix  $\Phi$ , but this time it is not the adjoint of the sampling operator  $\tilde{\Phi}^*$ . For a given input vector  $y \in \mathbb{C}^N$ , the interpolation output is a vector  $\hat{x} \in \mathbb{C}^M$ ;  $\Phi$  looks the same as in (4.13). When the interpolation operator is specially chosen so that

$$\Phi = \tilde{\Phi}(\tilde{\Phi}^* \tilde{\Phi})^{-1}, \quad (4.18)$$

that is, it is the pseudoinverse of  $\tilde{\Phi}$ , then  $\tilde{S} = S$ , because

$$S = \mathcal{R}(\Phi) \stackrel{(a)}{=} \mathcal{R}(\tilde{\Phi}(\tilde{\Phi}^* \tilde{\Phi})^{-1}) \stackrel{(b)}{=} \mathcal{R}(\tilde{\Phi}) \stackrel{(c)}{=} \mathcal{N}(\tilde{\Phi}^*)^\perp \stackrel{(d)}{=} \tilde{S}, \quad (4.19)$$

where (a) follows from (4.18); (b) because  $\mathcal{R}(AB) = \mathcal{R}(A)$ ; (c) from (1.49a); and (d) from the definition of  $\tilde{S}$ .

**EXAMPLE 4.5 (INTERPOLATION IN  $\mathbb{R}^4$ )** Define the interpolation operator as the pseudoinverse of (4.17a),

$$\Phi = \frac{1}{2} \begin{bmatrix} 3 & -2 & 1 \\ 1 & 2 & -1 \\ -1 & 2 & 1 \\ 1 & -2 & 3 \end{bmatrix}, \quad (4.20a)$$

illustrated in Figure 4.10(b). With  $\alpha_k \in \mathbb{R}$ , the range of  $\Phi$  is

$$S = \frac{1}{2} \left\{ \alpha_0 \begin{bmatrix} 3 \\ 1 \\ -1 \\ 1 \end{bmatrix} + \alpha_1 \begin{bmatrix} -2 \\ 2 \\ 2 \\ -2 \end{bmatrix} + \alpha_2 \begin{bmatrix} 1 \\ -1 \\ 1 \\ 3 \end{bmatrix} \right\}. \quad (4.20b)$$

By equating (4.17b) and (4.20b), we can easily check that  $\tilde{S} = S$ .

**Interpolation Followed by Sampling** Interpolation followed by sampling is described by  $\tilde{\Phi}^* \Phi$ , which maps from the smaller space,  $\mathbb{C}^N$ , to itself. From the dimensions of the operators, it is possible for  $y$  to be perfectly recovered; this happens when  $\tilde{\Phi}^*$  is the left inverse of  $\Phi$ ,

$$\tilde{\Phi}^* \Phi = I \quad \Leftrightarrow \quad \langle \varphi_n, \tilde{\varphi}_k \rangle = \delta_{n-k}. \quad (4.21)$$

The sampling and interpolation operators are then called *consistent*. Choosing the pseudoinverse in (4.18) for  $\Phi$  satisfies (4.21); of course, there exist infinitely many other left inverses we could also use. The above also shows that the condition for perfect recovery is the same as the sets of vectors  $\{\varphi_k\}_{k=0}^{N-1}$  and  $\{\tilde{\varphi}_k\}_{k=0}^{N-1}$  being biorthogonal, as in (1.102). These sets of vectors are not bases for  $\mathbb{C}^M$  (too few vectors); instead, they form a biorthogonal pair of bases for the  $N$ -dimensional subspaces  $S$  and  $\tilde{S}$  they span, respectively.

**EXAMPLE 4.6 (INTERPOLATION FOLLOWED BY SAMPLING IN  $\mathbb{R}^4$ )** Because we have chosen the interpolation operator in (4.20a) to be the pseudoinverse of the sampling operator in (4.17a), interpolation followed by sampling leads to perfect recovery,

$$\tilde{\Phi}^* \Phi = \frac{1}{2} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \frac{1}{2} \begin{bmatrix} 3 & -2 & 1 \\ 1 & 2 & -1 \\ -1 & 2 & 1 \\ 1 & -2 & 3 \end{bmatrix} = I. \quad (4.22)$$

**Sampling Followed by Interpolation** Sampling followed by interpolation is described by  $P = \Phi \tilde{\Phi}^*$ . When the sampling and interpolation operators are consistent as in (4.21), then

$$P^2 = \Phi \tilde{\Phi}^* \Phi \tilde{\Phi}^* \stackrel{(a)}{=} \Phi \tilde{\Phi}^* = P,$$

where (a) follows from consistency. In other words, the idempotency of  $P$  is guaranteed by consistency, that is, consistency implies that  $P$  is a projection operator, albeit not necessarily an orthogonal one.<sup>78</sup> Figure 4.11 shows what happens in that case;  $P$  projects onto  $S$ , but the projection is not orthogonal. The approximation error  $x - \hat{x}$  is orthogonal to  $\tilde{S}$  but not to  $S$ .

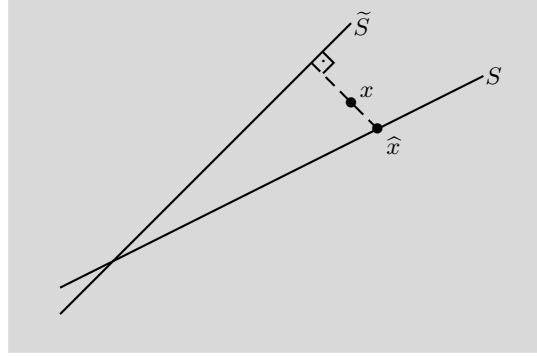
It turns out that for  $P$  to be self-adjoint as well,  $\Phi$  must be chosen to be the pseudoinverse of  $\tilde{\Phi}^*$ , (4.18); the sampling and interpolation operators are then called *ideally matched*,

$$\begin{aligned} P^* &= (\Phi \tilde{\Phi}^*)^* \stackrel{(a)}{=} (\tilde{\Phi}(\tilde{\Phi}^* \tilde{\Phi})^{-1} \tilde{\Phi}^*)^* = \tilde{\Phi}((\tilde{\Phi}^* \tilde{\Phi})^{-1})^* \tilde{\Phi}^* \\ &= \tilde{\Phi}(\tilde{\Phi}^* \tilde{\Phi})^{-1} \tilde{\Phi}^* \stackrel{(b)}{=} \Phi \tilde{\Phi}^* = P, \end{aligned}$$

where (a) and (b) follow from (4.18). In other words, the self-adjointness of  $P$  is guaranteed by sampling and interpolation being ideally matched and  $S = \tilde{S}$ .

The previous discussion can be summarized as follows (see also Figure 4.11):

<sup>78</sup>Projection operator and oblique projection operators are synonyms (see Chapter 1; we choose to use projection operator to mean a nonorthogonal projection operator).



**Figure 4.11:** Subspaces defined in sampling and interpolation.  $\tilde{S}$  represents what can be measured; it is the orthogonal complement of the null space of the sampling operator  $\tilde{\Phi}^*$ .  $S$  represents what can be reproduced; it is the range of the interpolation operator  $\Phi$ . When sampling and interpolation are *consistent*,  $\Phi\tilde{\Phi}^*$  is a projection and  $x - \hat{x}$  is orthogonal to  $\tilde{S}$ . When furthermore  $S = \tilde{S}$ , the projection becomes an orthogonal projection.

**THEOREM 4.2 (RECOVERY FOR FINITE-DIMENSIONAL VECTORS)** Given is sampling followed by interpolation with sampling operator  $\tilde{\Phi}^*$  from (4.16) and interpolation operator  $\Phi$  from (4.13). Then, with  $S = \mathcal{R}(\Phi)$  and  $\tilde{S} = \mathcal{N}(\tilde{\Phi}^*)^\perp$ :

- (i) When  $P = \Phi\tilde{\Phi}^*$  is idempotent, that is, sampling and interpolation operators are consistent, then  $P$  is a projection operator. When  $x \in S$ ,  $x$  is perfectly recovered.
- (ii) When  $P = \Phi\tilde{\Phi}^*$  is idempotent and self-adjoint, that is, sampling and interpolation operators are consistent and ideally matched, then  $P$  is an orthogonal projection operator. When  $x \in S$ ,  $x$  is perfectly recovered.

**EXAMPLE 4.7 (SAMPLING FOLLOWED BY INTERPOLATION IN  $\mathbb{R}^4$ )** The interpolation operator  $\Phi$  in (4.20a) is the pseudoinverse of the sampling operator  $\tilde{\Phi}^*$  in (4.17a); they are thus ideally matched, and  $P$  is self-adjoint. Together with consistency, (4.22), which guarantees that  $P$  is idempotent, we get that  $P$  is an orthogonal projection operator, and projects onto  $S$ ,

$$P = \Phi\tilde{\Phi}^* = \frac{1}{4} \begin{bmatrix} 3 & -2 & 1 \\ 1 & 2 & -1 \\ -1 & 2 & 1 \\ 1 & -2 & 3 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 3 & 1 & -1 & 1 \\ 1 & 3 & 1 & -1 \\ -1 & 1 & 3 & 1 \\ 1 & -1 & 1 & 3 \end{bmatrix}.$$

Take  $x = [2 \ 0 \ 0 \ 2]^T \in S$ . A simple calculation shows that  $Px = x$ , that is, perfect recovery. Choosing a vector not in  $S$ , on the other hand, would result in a best least-squares approximation of  $x$  in  $S$ . If  $[1 \ -1 \ 1 \ 0]^T$ , then  $Px = 0$ .



To get ideally-matched sampling and interpolation operators that are adjoints of each other, we can apply Gram–Schmidt orthogonalization to  $\tilde{\Phi}$  (see Algorithm 1.1), yielding a new pair of sampling and interpolation operators,  $\Psi^*$  and  $\Psi$ , this time with orthogonal vectors,

$$\Psi = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} & \frac{1}{2\sqrt{3}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & -\frac{1}{2\sqrt{3}} \\ 0 & \frac{2}{\sqrt{6}} & \frac{2}{\sqrt{3}} \\ 0 & 0 & \frac{3}{2\sqrt{3}} \end{bmatrix}.$$

### 4.3 Sequences

Now that we have a firm grasp of the matrix view of sampling and interpolation, we can move to spaces with domains associated with time and restrict the sampling and interpolation operators accordingly. In these developments, we emphasize the cases where ideally-matched sampling and interpolation operators are adjoints of each other, as they are well-suited to implementations using LSI filtering.

**Shift-Invariant Subspaces of Sequences** We start by introducing a class of subspaces of  $\ell^2(\mathbb{Z})$  that will play a prominent role in the material that follows.

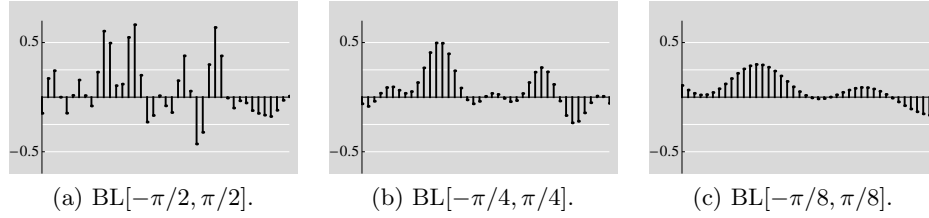
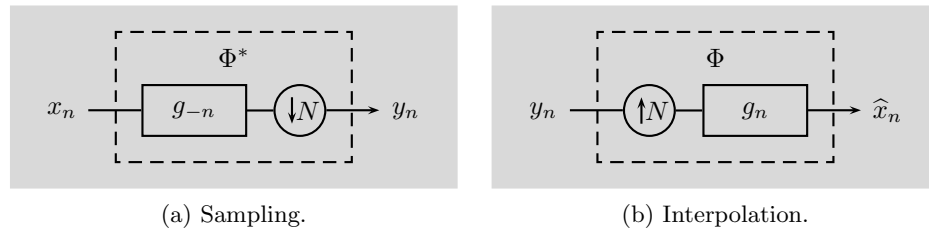
**DEFINITION 4.3 (SHIFT-INVARIANT SUBSPACES OF  $\ell^2(\mathbb{Z})$ )** A subspace  $W \subset \ell^2(\mathbb{Z})$  is a *shift-invariant subspace* with respect to shift  $L \in \mathbb{Z}^+$  when  $x_n \in W$  implies  $x_{n-kL} \in W$  for every integer  $k$ . In addition,  $w \in \ell^2(\mathbb{Z})$  is called a *generator* of  $W$  when  $W = \overline{\text{span}}(\{w_{n-kL}\}_{k \in \mathbb{Z}})$ .

For example, the outputs of upsampling followed by filtering, (2.198), form a shift-invariant subspace with respect to integer multiples of 2: A shift of  $\hat{x}_n$  by  $2k$  is still in the same subspace. We will see that the same will be true for the sampling and interpolation operators we define shortly.

**Subspaces of Bandlimited Sequences** A special case of shift-invariant subspaces of particular importance in signal processing is the subspace of bandlimited sequences; we now define it formally and later we look at forming approximations in these subspaces through sampling and interpolation.

**DEFINITION 4.4 (BANDWIDTH FOR SEQUENCES)** A sequence  $x_n \in \ell^2(\mathbb{Z})$  is said to have *bandwidth*  $\omega_0$  for the smallest  $\omega_0 \in (0, 2\pi]$  such that the discrete-time Fourier transform  $X(e^{j\omega})$  satisfies

$$X(e^{j\omega}) = 0 \quad \text{for all } |\omega| > \frac{\omega_0}{2}. \quad (4.23)$$

**Figure 4.12:** Sequences in  $\text{BL}[-\pi/N, \pi/N]$  subspaces.**Figure 4.13:** Sampling and interpolation in  $\ell^2(\mathbb{Z})$  with orthonormal sequences.

**DEFINITION 4.5 (SUBSPACE OF BANDLIMITED SEQUENCES)** A subspace  $\text{BL}[-\omega_0/2, \omega_0/2] \subset \ell^2(\mathbb{Z})$  is a *subspace of bandlimited sequences* when all  $x_n \in \text{BL}[-\omega_0/2, \omega_0/2]$  have bandwidth at most  $\omega_0$ .

A subspace of bandlimited sequences is shift invariant for any shift  $L \in \mathbb{Z}^+$ ; in fact, subspaces of bandlimited sequences are the only ones that are simultaneously shift invariant for all shifts  $L \in \mathbb{Z}^+$ . To see shift invariance, take  $x_n \in \text{BL}[-\omega_0/2, \omega_0/2]$ . Then, (2.85) states that

$$x_{n-kL} \xrightarrow{\text{DTFT}} e^{-j\omega kL} X(e^{j\omega});$$

the DTFT is multiplied by a complex exponential, not changing the bandwidth of the shifted sequence. Figure 4.12 illustrates a few of such subspaces.

### 4.3.1 Sampling and Interpolation with Orthonormal Sequences

As we have done for finite-dimensional vectors, we start with the case when the relevant spaces are spanned by orthonormal sets. This case, apart from intuitive geometric properties, is both simpler, and better known in practice.

**Sampling** We refer to the operation depicted in Figure 4.13(a), involving filtering with  $g_{-n}$  and downsampling by integer  $N > 1$ , as *sampling of the sequence*  $x_n \in \ell^2(\mathbb{Z})$  with *prefilter*  $g_{-n}$ , and denote it by  $y_n = (\Phi^* x)_n$ . Even though it results in an infinite number of samples, it involves a dimensionality reduction because there

## 4.3. Sequences

395

is only one output sample per  $N$  input samples. We have seen this combination for  $N = 2$  in Section 2.7.4 and the expression for the output in (2.195).

Generalizing (2.195) for an arbitrary  $N$ , the output of sampling is

$$\begin{aligned} y_n &= (g_{-n} * x_n)|_{nN} = \sum_{k \in \mathbb{Z}} g_{-k} x_{Nn-k} = \langle x_{Nn-k}, g_{-k} \rangle_k \\ &= \sum_{k \in \mathbb{Z}} x_k g_{k-Nn} = \langle g_{k-Nn}, x_k \rangle_k \\ &\stackrel{(a)}{=} \langle x_k, \varphi_{k-Nn} \rangle_k = (\Phi^* x)_n, \end{aligned} \quad (4.24)$$

where (a) follows from  $\varphi_n = g_n$ . The sampling operator  $\Phi^*$  is now an infinite matrix (see (2.194) for  $N = 2$ ), with rows equal to  $\varphi^*$  and its shifts by integer multiples of  $N$ . We assume these rows to be orthonormal,

$$\langle \varphi_{n-N\ell}, \varphi_{n-Nk} \rangle = \delta_{\ell-k} \quad \Leftrightarrow \quad \Phi^* \Phi = I. \quad (4.25)$$

As before, the sampling operator has a nontrivial null space,  $S^\perp = \mathcal{N}(\Phi^*)$ ; the set  $\{\varphi_k\}_{k \in \mathbb{Z}}$  spans its orthogonal complement,  $S = \mathcal{N}(\Phi^*)^\perp = \text{span}(\{\varphi_k\}_{k \in \mathbb{Z}})$ . In other words, when a sequence  $x_n \in \ell^2(\mathbb{Z})$  is sampled, the component that remains is in  $S$  and is captured by  $\Phi^* x$ ; the component that is lost due to sampling is in the null space  $S^\perp$ . We illustrate this with an example.

EXAMPLE 4.8 (SAMPLING IN  $\ell^2(\mathbb{Z})$ ) Choose  $N = 2$  and the prefilter to be

$$g_{-n} = \frac{1}{\sqrt{2}} \begin{bmatrix} \dots & 0 & 1 & \boxed{1} & 0 & 0 & \dots \end{bmatrix}. \quad (4.26)$$

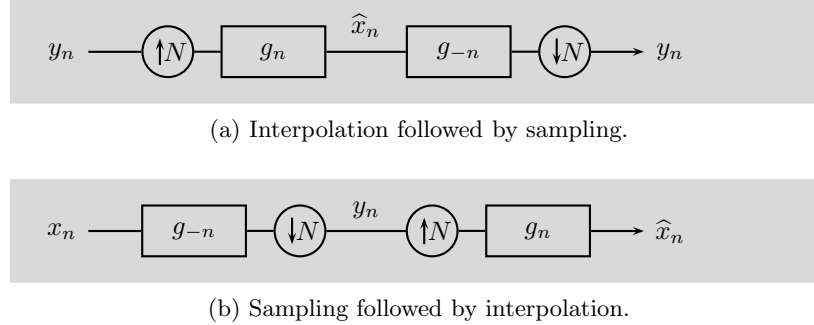
Then from (2.194), the output is

$$\begin{bmatrix} \vdots \\ \boxed{y_0} \\ y_1 \\ \vdots \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & \boxed{1} & 1 & 0 & 0 & \dots \\ \dots & 0 & 0 & 1 & 1 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ \boxed{x_0} \\ x_1 \\ x_2 \\ x_3 \\ \vdots \end{bmatrix} = \Phi^* x. \quad (4.27a)$$

Clearly then, for every two input samples,  $x_{2k}$  and  $x_{2k+1}$ , we get one output sample,  $y_k = (x_{2k} + x_{2k+1})/\sqrt{2}$ . Also, it is obvious that the matrix  $\Phi^*$  has orthonormal rows, since their nonzero parts do not overlap. With  $\alpha_k \in \mathbb{R}$ , the null space of  $\Phi^*$  and its orthogonal complement are

$$\begin{aligned} \mathcal{N}(\Phi^*) &= \{x \in \ell^2(\mathbb{Z}) \mid x_{2k} = -x_{2k+1}, k \in \mathbb{Z}\}, \\ S &= \{\alpha_k g_{n-2k}\}_{k \in \mathbb{Z}}. \end{aligned} \quad (4.27b)$$

Then any  $x_n \in \ell^2(\mathbb{Z})$  can be decomposed into a part belonging to the null space  $S^\perp = \mathcal{N}(\Phi^*)$ , which will be lost during sampling, and to a second part belonging to  $S$ , which will be preserved during sampling.



**Figure 4.14:** Sampling and interpolation in  $\ell^2(\mathbb{Z})$  with orthonormal sequences.

**Interpolation** We refer to the operation depicted in Figure 4.13(b), involving up-sampling by integer  $N > 1$  and filtering with  $g_n$ , as *interpolation of the sequence*  $y_n \in \ell^2(\mathbb{Z})$  with *postfilter*  $g_n$ , and denote it by  $\hat{x}_n = (\Phi y)_n$ . We have seen this combination for  $N = 2$  in Section 2.7.4 and the expression for the output in (2.198). The way we chose pre- and postfilters, the sampling and interpolation operators are adjoints of each other.

Generalizing (2.198) for an arbitrary  $N$ , the output of interpolation is

$$\begin{aligned} \hat{x}_n &= \sum_{k \in \mathbb{Z}} y_k g_{n-Nk} = \langle y_k, g_{n-Nk} \rangle_k \\ &\stackrel{(a)}{=} \langle y_k, \varphi_{n-Nk} \rangle_k = (\Phi y)_n, \end{aligned} \quad (4.28)$$

where (a) follows from  $\varphi_n = g_n$ . The interpolation operator  $\Phi$  is also an infinite matrix (see (2.197) for  $N = 2$ ), with columns equal to  $\varphi$  and its shifts by integer multiples of  $N$ . Denoting the range of  $\Phi$  by  $S$  as before, this subspace is, of course, the same as the orthogonal complement of the null space of the sampling operator, as we have seen earlier. It is also a shift-invariant subspace with respect to integer multiples of  $N$ : A shift of  $\hat{x}_n$  by  $kN$  is still in  $S$ .

**EXAMPLE 4.9 (INTERPOLATION IN  $\ell^2(\mathbb{Z})$ )** From (4.27a), the output of interpolation is

$$\begin{bmatrix} \vdots \\ \hat{x}_0 \\ \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \vdots \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} \ddots & \vdots & \vdots & \ddots \\ \dots & 1 & 0 & \dots \\ \dots & 1 & 0 & \dots \\ \dots & 0 & 1 & \dots \\ \dots & 0 & 1 & \dots \\ \ddots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ y_0 \\ y_1 \\ \vdots \end{bmatrix} = \Phi y. \quad (4.29)$$

Clearly then, for every input sample,  $y_k$ , we get two output samples  $x_{2k} = x_{2k+1} = y_k/\sqrt{2}$ . The range of this operator is  $S$  (the same as the orthogonal complement of the null space of the sampling operator in (4.27b)).

**Interpolation Followed by Sampling** Interpolation followed by sampling is described by  $\Phi^*\Phi$  as in Figure 4.14(a). Since by assumption (4.25) holds,  $y_n$  is perfectly recovered. Equation (4.25) also shows that the condition for perfect recovery is the same as the set of sequences  $\{\varphi_{n-Nk}\}_{k \in \mathbb{Z}}$  being orthonormal, as in (1.83). This set of sequences is not a basis for  $\ell^2(\mathbb{Z})$ ; instead, it is an orthonormal basis for the subspace  $S$  it spans.

**Sampling Followed by Interpolation** Sampling followed by interpolation is described by  $P = \Phi\Phi^*$  as in Figure 4.14(b). We know this to be a more difficult sequence of operations to recover from perfectly, as sampling, unless the input is in  $S$ , leads to a loss.

As for finite-dimensional vectors, and given our choice of sampling and interpolation operators,

$$\begin{aligned} P^2 &= \Phi\Phi^*\Phi\Phi^* \stackrel{(a)}{=} \Phi\Phi^* = P, \\ P^* &= (\Phi\Phi^*)^* = \Phi\Phi^* = P, \end{aligned}$$

where (a) follows from (4.25). In other words,  $P$  is idempotent and self-adjoint, that is,  $P$  is an orthogonal projection operator. Then, by Theorem 1.26,  $(Px)_n$  is the best least-squares approximation of  $x_n$  in  $S$ ; if  $x_n \in S$ , sampling followed by interpolation will perfectly recover  $x_n$ :

**THEOREM 4.6 (RECOVERY FOR SEQUENCES)** Given is the system as in Figure 4.14(b) with sampling operator  $\Phi^*$  from (4.24) and interpolation operator  $\Phi$  satisfying (4.25). Then, with  $S = \mathcal{R}(\Phi)$ ,

$$\hat{x}_n = (\Phi\Phi^*x)_n \tag{4.30}$$

is the best least-squares approximation of  $x_n$  in  $S$ , that is,

$$\hat{x}_n = \min_{x_{S,n} \in S} \|x_n - x_{S,n}\|^2, \quad \hat{x}_n - x_n \perp S.$$

When  $x_n \in S$ , then  $\hat{x}_n = x_n$ .

**EXAMPLE 4.10** We now illustrate both theorems.

Choose first  $x_n \in S$ , a piecewise-constant sequence over intervals of length 2,

$$x = [\dots \quad x_{-2} \quad x_{-2} \quad \boxed{x_0} \quad x_0 \quad x_2 \quad x_2 \quad \dots].$$

Then the results of applying filtering by  $g_{-n}$  from (4.26), downsampling by 2,

upsampling by 2 and filtering by  $g_n$  as in Figure 4.14(b) are, respectively,

$$\begin{aligned} g_{-n} * x_n &= \begin{bmatrix} \dots & \boxed{\sqrt{2}x_0} & \frac{1}{\sqrt{2}}(x_0 + x_2) & \sqrt{2}x_2 & \frac{1}{\sqrt{2}}(x_2 + x_4) & \dots \end{bmatrix} \\ y_n &= \begin{bmatrix} \dots & \boxed{\sqrt{2}x_0} & & \sqrt{2}x_2 & & \dots \end{bmatrix}, \\ y_{n/2} &= \begin{bmatrix} \dots & \boxed{\sqrt{2}x_0} & 0 & \sqrt{2}x_2 & 0 & \dots \end{bmatrix} \\ \hat{x}_n = x &= \begin{bmatrix} \dots & \boxed{x_0} & x_0 & x_2 & x_2 & \dots \end{bmatrix} \end{aligned}$$

that is, perfect recovery of  $x_n$ .

Choose next  $x \notin S$ ,

$$x = \begin{bmatrix} \dots & 0 & \boxed{2} & 0 & 0 & 2 & 0 & \dots \end{bmatrix}.$$

The best we can do now is compute an approximation. Applying  $P = \Phi\Phi^*$ ,

$$\begin{aligned} g_{-n} * x_n &= \begin{bmatrix} \dots & \sqrt{2} & \boxed{\sqrt{2}} & 0 & \sqrt{2} & \sqrt{2} & 0 & \dots \end{bmatrix} \\ y_n &= \begin{bmatrix} \dots & & \boxed{\sqrt{2}} & & \sqrt{2} & & 0 & \dots \end{bmatrix}, \\ y_{n/2} &= \begin{bmatrix} \dots & 0 & \boxed{\sqrt{2}} & 0 & \sqrt{2} & 0 & 0 & \dots \end{bmatrix} \\ \hat{x}_n &= \begin{bmatrix} \dots & 0 & \boxed{1} & 1 & 1 & 1 & 0 & \dots \end{bmatrix} \end{aligned}$$

which clearly belongs to  $S$  since it is piecewise constant over intervals of length 2. It is also the closest sequence in  $S$ , since the error between  $x_n$  and  $\hat{x}_n$  is

$$x_n - \hat{x}_n = \begin{bmatrix} \dots & 0 & 1 & -1 & -1 & 1 & 0 & \dots \end{bmatrix}^T \perp S.$$

### 4.3.2 Sampling and Interpolation for Bandlimited Sequences

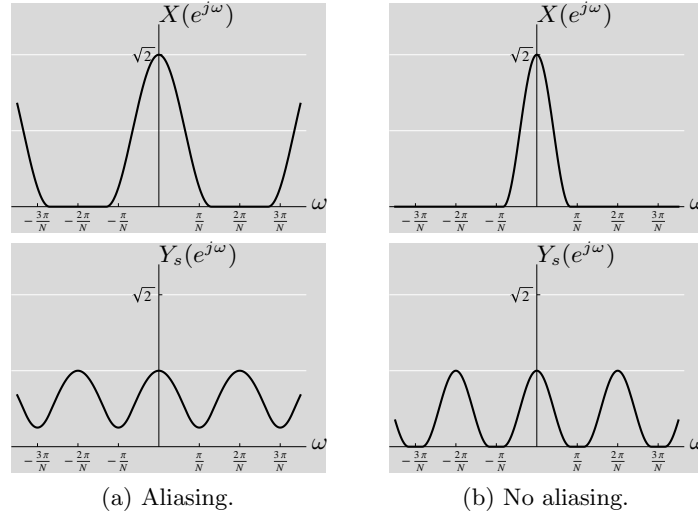
Since a subspace of bandlimited sequences is also a shift-invariant subspace, everything we have seen thus far holds here as well. In particular, if  $x_n \in \text{BL}[-\omega_0/2, \omega_0/2]$ , sampling operator  $\Phi^*$  is given by (4.24) and interpolation operator is  $\Phi$ , then by Theorem 4.6,  $x_n$  is perfectly recovered after sampling followed by interpolation. We see that the requirement  $x_n \in \text{BL}[-\omega_0/2, \omega_0/2]$  is more restrictive than  $x_n \in S$ . If, on the other hand,  $x_n \notin \text{BL}[-\omega_0/2, \omega_0/2]$ , then by Theorem 4.6,  $P = \Phi\Phi^*$  will project onto  $S$ , which is not necessarily equal to  $\text{BL}[-\omega_0/2, \omega_0/2]$ .

For  $P$  to orthogonally project onto  $\text{BL}[-\omega_0/2, \omega_0/2]$ , we clearly need  $P$  to be a bit more specific. We now establish what the sampling and interpolation operators must be using the machinery from Chapter 2.

Assume first that  $x_n \in \text{BL}[-\omega_0/2, \omega_0/2]$  and that the sampling prefilter in Figure 4.14(b) is a simple multiplicative factor of  $\sqrt{N}$ .<sup>79</sup> Then the output after upsampling by  $N$  is

$$Y_s(e^{j\omega}) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X(e^{j(\omega - 2\pi k/N)}). \quad (4.31)$$

<sup>79</sup>Basically, the sampling prefilter is not present; multiplication by  $\sqrt{N}$  is purely for convenience.



**Figure 4.15:** Downsampling by  $N$  followed by upsampling by  $N$  of  $x_n \in \text{BL}[-\omega_0/2, \omega_0/2]$ . (a) When  $\omega_0 > 2\pi/N$ , spectral replicas overlap;  $x_n$  cannot be recovered by LSI filtering for every  $x_n \in \text{BL}[-\omega_0/2, \omega_0/2]$ . (b) When  $\omega_0 \leq 2\pi/N$ , spectral replicas do not overlap;  $x_n$  can be recovered by lowpass filtering by  $g_n$ . (Illustrated for  $N = 4$ .)

For some LSI postfilter  $g_n$  to recover  $x_n$ , a pointwise multiplication of (4.31) by  $G(e^{j\omega})$  must yield  $X(e^{j\omega})$ . We want the recovery to work for every  $x_n \in \text{BL}[-\omega_0/2, \omega_0/2]$ , so we cannot count on any property of  $X(e^{j\omega})$  other than bandlimitedness (4.23). Thus, multiplication by  $G(e^{j\omega})$  must compensate for the  $1/\sqrt{N}$  factor in the  $k = 0$  term of (4.31) and zero out all the other terms.

Whether the multiplication by  $G(e^{j\omega})$  will recover  $X(e^{j\omega})$  depends on whether the spectral replicas in (4.31) overlap. Figure 4.15 illustrates the two possibilities, which we now discuss in more detail.

**Aliasing** When  $\omega_0 > 2\pi/N$  as in Figure 4.15(a), spectral replicas overlap, and no LSI filtering will succeed in recovering  $x_n$  for every  $x_n \in \text{BL}[-\omega_0/2, \omega_0/2]$ . This confusion of frequencies is called *aliasing*; it is one of the most well-known effects of sampling in general and is the reason why a prefilter is needed before downsampling. We will discuss aliasing in more detail in the next section.

**Sampling Theorem** When  $\omega_0 \leq 2\pi/N$  as in Figure 4.15(b), spectral replicas do not overlap, and obtaining  $\hat{x}_n = x_n$  requires  $G(e^{j\omega})$  to be an ideal filter with cut-off frequency  $\omega_0/2$ . Choosing exactly  $\omega_0 = 2\pi/N$  uniquely determines the postfilter as

$$G(e^{j\omega}) = \begin{cases} \sqrt{N}, & |\omega| \leq \pi/N; \\ 0, & \text{otherwise,} \end{cases} \quad \xleftrightarrow{\text{DTFT}} \quad g_n = \frac{1}{\sqrt{N}} \text{sinc}\left(\frac{\pi}{N}n\right), \quad (4.32)$$

an ideal  $N$ th-band filter from Table 2.5. Intuitively, this tells us that  $x_n$  can be recovered exactly after keeping  $1/N$ th of its samples only when it occupies less than

1/ $N$ th of the full band in the DTFT domain. Since  $\{g_{n-Nk}\}_{k \in \mathbb{Z}}$  are orthonormal, we can choose the sampling prefilter to be  $g_{-n}$ ,<sup>80</sup> leading to an orthogonal projection operator  $P$ . Because  $x_n \in \text{BL}[-\pi/N, \pi/N]$ , this sampling prefilter will have no effect on  $x_n$ . By construction, the orthonormal set  $\{g_{n-Nk}\}_{k \in \mathbb{Z}}$  is an orthonormal basis for  $\text{BL}[-\omega_0/2, \omega_0/2]$ .

This discussion leads us to the sampling theorem for sequences:

**THEOREM 4.7 (SAMPLING THEOREM FOR SEQUENCES)** Given is the system as in Figure 4.14(b) with interpolation postfilter  $g_n$  from (4.32). Then,

$$x_n = \sum_{k \in \mathbb{Z}} x_k \text{sinc}\left(\frac{\pi}{N}(n - kN)\right) \quad \Leftrightarrow \quad x_n \in \text{BL}\left[-\frac{\pi}{N}, \frac{\pi}{N}\right]. \quad (4.33)$$

The expression (4.33) comes from

$$\begin{aligned} x_n &= (\Phi\Phi^*x)_n \stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} \langle x_\ell, g_{\ell-Nk} \rangle_\ell g_{n-Nk} \\ &\stackrel{(b)}{=} \frac{1}{N} \sum_{k \in \mathbb{Z}} \langle x_\ell, \text{sinc}\left(\frac{\pi}{N}(\ell - kN)\right) \rangle_\ell \text{sinc}\left(\frac{\pi}{N}(n - kN)\right) \\ &\stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} x_k \text{sinc}\left(\frac{\pi}{N}(n - kN)\right), \end{aligned}$$

where (a) follows from (4.24) and (4.28); (b) from  $g_n = (1/\sqrt{N}) \text{sinc}(\pi n/N)$ ; and (c) from  $x_n \in \text{BL}[-\pi/N, \pi/N]$  and thus the effect of  $\text{sinc}((\pi/N)(n - kN))$  on  $x_n$  is just multiplication by  $N$ . This theorem can also be seen as a corollary of Theorem 4.6.

**Bandlimited Approximation of Sequences** We now assume that  $x \notin \text{BL}[-\pi/N, \pi/N]$ . Then, as a corollary to Theorem 4.6, since  $P$  is an orthogonal projection operator and  $S = \text{BL}[-\pi/N, \pi/N]$ :

**THEOREM 4.8 (BEST LEAST-SQUARES BANDLIMITED APPROXIMATION)** Given is the system as in Figure 4.14(b) with interpolation postfilter  $g_n$  from (4.32). Then,

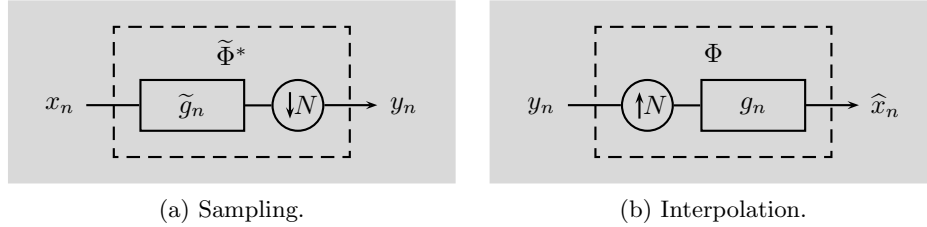
$$\hat{x}_n = (\Phi\Phi^*x)_n \quad (4.34)$$

is the best least-squares approximation of  $x_n$  in  $\text{BL}[-\pi/N, \pi/N]$ , that is,

$$\hat{x}_n = \min_{x_{\text{BL},n} \in \text{BL}[-\pi/N, \pi/N]} \|x_n - x_{\text{BL},n}\|^2, \quad \hat{x}_n - x_n \perp \text{BL}[-\pi/N, \pi/N].$$

<sup>80</sup>We choose a time-reversed version to yield an orthogonal projection operator  $P$ . This time reversal has no effect on the ideal filter since its impulse response is symmetric.





**Figure 4.16:** Sampling and interpolation in  $\ell^2(\mathbb{Z})$  with nonorthogonal sequences.

The effect of this approximation in the DTFT domain is a simple truncation of the spectrum of  $x_n$  to  $[-\pi/N, \pi/N]$ :

$$\hat{X}(e^{j\omega}) = \begin{cases} X(e^{j\omega}), & |\omega| \leq \pi/N; \\ 0, & \text{otherwise.} \end{cases}$$

Exercise 4.6 explores bandlimited spaces with rational sampling rate changes.

### 4.3.3 Sampling and Interpolation with Nonorthogonal Sequences

As for finite-dimensional vectors, what we have seen thus far is a classical take on sampling and interpolation; we now expand it a bit to include nonorthogonal sequences. These make the geometry more complicated; the sampling and interpolation operators are no longer adjoints of each other, and the appropriate spaces we discussed earlier, the range of the interpolation operator as well as the orthogonal complement of the null space of the sampling operator, are no longer the same.

**Sampling** We now refer to the operation depicted in Figure 4.16(a), involving filtering with  $\tilde{g}_n$  and downsampling by integer  $N > 1$ , as *sampling of the sequence*  $x_n \in \ell^2(\mathbb{Z})$  with prefilter  $\tilde{g}_n$ , and denote it by  $y_n = (\tilde{\Phi}^* x)_n$ . This time, we do not make an assumption of orthonormality.

As in (4.24), we generalize (2.195) for an arbitrary  $N$ ,

$$\begin{aligned} y_n &= (\tilde{g} * x)|_{nN} = \sum_{k \in \mathbb{Z}} \tilde{g}_k x_{Nn-k} = \langle x_{Nn-k}, \tilde{g}_k \rangle_k \\ &= \sum_{k \in \mathbb{Z}} x_k \tilde{g}_{Nn-k} = \langle \tilde{g}_{Nn-k}, x_k \rangle_k \\ &\stackrel{(a)}{=} \langle x_k, \tilde{\varphi}_{k-Nn} \rangle_k = (\tilde{\Phi}^* x)_n, \end{aligned} \tag{4.35}$$

where (a) follows from  $\tilde{\varphi}_n = \tilde{g}_{-n}$ . The sampling operator  $\tilde{\Phi}^*$  is again an infinite matrix (see (2.194) for  $N = 2$ ), with rows equal to  $\tilde{\varphi}^*$  and its shifts by integer multiples of  $N$ . The null space of  $\tilde{\Phi}^*$  is nontrivial (see Solved Exercise 4.3). The orthogonal complement of this null space is denoted  $\tilde{S}$  as before.

EXAMPLE 4.11 (SAMPLING IN  $\ell^2(\mathbb{Z})$ ) Choose  $N = 2$  and the prefilter to be

$$\tilde{g}_n = \frac{1}{8} [\dots \ 0 \ -1 \ 2 \ \boxed{6} \ 2 \ -1 \ 0 \ \dots]. \quad (4.36)$$

Then from (2.194), the output is

$$\begin{bmatrix} \vdots \\ y_0 \\ y_1 \\ \vdots \end{bmatrix} = \frac{1}{8} \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & -1 & 2 & \boxed{6} & 2 & -1 & 0 & 0 & \dots \\ \dots & 0 & 0 & -1 & 2 & 6 & 2 & -1 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ x_0 \\ x_1 \\ x_2 \\ x_3 \\ \vdots \end{bmatrix} = \tilde{\Phi}^* x. \quad (4.37a)$$

Again, for every two input samples we get one output sample. Also, it is obvious that the matrix  $\tilde{\Phi}^*$  does not have orthogonal rows. With  $\alpha_k \in \mathbb{R}$ , the null space of  $\tilde{\Phi}^*$  and its orthogonal complement are

$$\begin{aligned} \mathcal{N}(\tilde{\Phi}^*) &= \{x \in \ell^2(\mathbb{Z}) \mid -x_{2k-2} + 2x_{2k-1} + 6x_{2k} + 2x_{2k+1} - x_{2k+2} = 0, k \in \mathbb{Z}\}, \\ \tilde{S} &= \{\alpha_k g_{2k-n}\}_{k \in \mathbb{Z}}. \end{aligned} \quad (4.37b)$$

Then any  $x_n \in \ell^2(\mathbb{Z})$  can be decomposed into a part belonging to the null space  $\tilde{S}^\perp = \mathcal{N}(\tilde{\Phi}^*)$ , which will be lost during sampling, and to a second part belonging to  $\tilde{S}$ , which will be preserved during sampling.

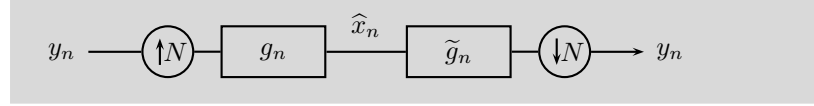
**Interpolation** Again, we refer to the operation depicted in Figure 4.16(b), involving upsampling by integer  $N > 1$  and filtering with  $g_n$ , as *interpolation of the sequence*  $y_n \in \ell^2(\mathbb{Z})$  with postfilter  $g_n$ , and denote it by  $\hat{x}_n = (\Phi y)_n$ , but this time it is not the adjoint of the sampling operator  $\tilde{\Phi}^*$ . It is, however, formally the same as in Figure 4.13(b) and (4.28). When the interpolation operator is specially chosen so that it formally satisfies (4.18), that is, it is the pseudoinverse of  $\tilde{\Phi}$ , then  $\tilde{S} = S$ , by the same arguments as in (4.19).

Both  $S$ , the range of the interpolation operator  $\Phi$  from (4.28), as well as  $\tilde{S}$ , the orthogonal complement of the null space of the sampling operator  $\tilde{\Phi}^*$  from (4.35), are shift-invariant subspaces with respect to integer multiples of  $N$  (see Exercise 4.3).

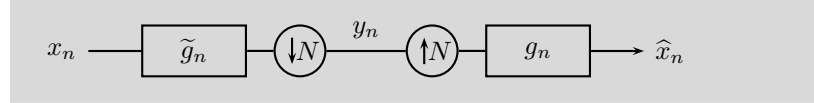
**Interpolation Followed by Sampling** Interpolation followed by sampling is described by  $\tilde{\Phi}^* \Phi$ , as in Figure 4.17(a). It is possible for  $y_n$  to be perfectly recovered; this happens when

$$\tilde{\Phi}^* \Phi = I \quad \Leftrightarrow \quad \langle \varphi_{n-N\ell}, \tilde{\varphi}_{n-Nk} \rangle = \delta_{\ell-k}. \quad (4.38)$$

The sampling and interpolation operators are then called *consistent*. Choosing the pseudoinverse in (4.18) for  $\Phi$  would satisfy (4.21); of course, there exist infinitely



(a) Interpolation followed by sampling.



(b) Sampling followed by interpolation.

**Figure 4.17:** Sampling and interpolation in  $\ell^2(\mathbb{Z})$  with nonorthogonal sequences.

many other  $\Phi$  we could also use. The above also shows that the condition for perfect recovery is the same as the sets of sequences  $\{\varphi_{n-Nk}\}_{k \in \mathbb{Z}}$  and  $\{\tilde{\varphi}_{n-Nk}\}_{k \in \mathbb{Z}}$  being biorthogonal, as in (1.102). These sets of sequences are not bases for  $\ell^2(\mathbb{Z})$ ; instead, they form a biorthogonal pair of bases for the subspaces  $S$  and  $\tilde{S}$  they span, respectively.

**EXAMPLE 4.12 (INTERPOLATION FOLLOWED BY SAMPLING IN  $\ell^2(\mathbb{Z})$ )** We start with  $N = 2$ , and assume  $\tilde{g}_n$  to be as in (4.36). We would like to find an interpolation operator such that (4.38) is satisfied. Assume for a moment that our interpolation prefilter  $g_n$  is of the following form:

$$g_n = a\delta_{n+1} + b\delta_n + a\delta_{n-1}.$$

To satisfy (4.38), we get the following system of equations:

$$2a + 3b = 4, \quad 2a - b = 0,$$

with the solution  $a = 1/2$ ,  $b = 1$ , or

$$g_n = \frac{1}{2} \begin{bmatrix} \dots & 0 & 1 & \boxed{2} & 1 & 0 & \dots \end{bmatrix}, \quad (4.39)$$

and the operator  $\Phi$  an infinite matrix with  $g_{n-2k}$  as its columns. With  $\alpha_k \in \mathbb{R}$ , the range of  $\Phi$  is

$$S = \{\alpha_k g_{n-2k}\}_{k \in \mathbb{Z}}. \quad (4.40)$$

**Sampling Followed by Interpolation** Sampling followed by interpolation is described by  $P = \Phi\tilde{\Phi}^*$ , as in Figure 4.17(b). When the sampling and interpolation operators are consistent as in (4.38), then  $P$  is idempotent, meaning it is a projection operator. It projects onto  $S$ , but the projection is not orthogonal. The approximation error  $x_n - \hat{x}_n$  is orthogonal to  $\tilde{S}$  but not to  $S$  (recall Figure 4.11 for a conceptual picture).

Again, for  $P$  to be self-adjoint as well,  $\Phi$  must be chosen to be the pseudoinverse of  $\tilde{\Phi}^*$ , (4.18); the sampling and interpolation operators are then *ideally matched*, and subspaces  $S$  and  $\tilde{S}$  are identical.

The previous discussion can be summarized as follows:

**THEOREM 4.9 (RECOVERY FOR SEQUENCES)** Given is the system as in Figure 4.17(b) with sampling operator  $\tilde{\Phi}^*$  from (4.35) and interpolation operator  $\Phi$  from (4.28). Then, with  $S = \mathcal{R}(\Phi)$  and  $\tilde{S} = \mathcal{N}(\tilde{\Phi}^*)^\perp$ :

- (i) When  $P = \Phi\tilde{\Phi}^*$  is idempotent, that is, sampling and interpolation operators are consistent, then  $P$  is a projection operator. When  $x_n \in S$ ,  $x_n$  is perfectly recovered.
- (ii) When  $P = \Phi\tilde{\Phi}^*$  is idempotent and self-adjoint, that is, sampling and interpolation operators are consistent and ideally matched, then  $P$  is an orthogonal projection operator. When  $x_n \in S$ ,  $x_n$  is perfectly recovered.

**EXAMPLE 4.13 (SAMPLING FOLLOWED BY INTERPOLATION IN  $\ell^2(\mathbb{Z})$ )** The interpolation operator  $\Phi$  in Example 4.12 is not the pseudoinverse of the sampling operator  $\tilde{\Phi}^*$  in (4.37a) (Exercise 4.8 does that); they are thus not ideally matched, and  $P$  is not self-adjoint,

$$P = \Phi\tilde{\Phi}^* = \frac{1}{16} \begin{bmatrix} \ddots & & & & & & & & & \\ \dots & -2 & 4 & 12 & 4 & -2 & 0 & 0 & 0 & \dots \\ \dots & 0 & 2 & 5 & 4 & 5 & 2 & 0 & 0 & \dots \\ \dots & 0 & 0 & -2 & 4 & 12 & 4 & -2 & 0 & \dots \\ \dots & 0 & 0 & 0 & 2 & 5 & 4 & 5 & 2 & \dots \\ \ddots & & & & & & & & & \ddots \end{bmatrix};$$

$P$  is consistent only, meaning it is a projection operator. The subspaces  $S$  and  $\tilde{S}$  are not the same.

## 4.4 Functions

In this section, we study sampling and interpolation operators that map between the continuous domain,  $\mathcal{L}^2(\mathbb{R})$ , and the discrete one,  $\ell^2(\mathbb{Z})$ . As before, we concentrate on structured operators, those that can be implemented using LSI filtering. Our development closely parallels the discrete-time case in the previous section.

**Shift-Invariant Subspaces of Functions** We start by introducing a class of subspaces of  $\mathcal{L}^2(\mathbb{R})$  that will play a prominent role in the material that follows.

**DEFINITION 4.10 (SHIFT-INVARIANT SUBSPACES OF  $\mathcal{L}^2(\mathbb{R})$ )** A subspace  $W \subset \mathcal{L}^2(\mathbb{R})$  is a *shift-invariant subspace* with respect to shift  $\tau \in \mathbb{R}^+$  when  $x(t) \in W$  implies  $x(t - k\tau) \in W$  for every integer  $k$ . In addition,  $w \in \mathcal{L}^2(\mathbb{R})$  is called a *generator* of  $W$  when  $W = \overline{\text{span}}(\{w(t - k\tau)\}_{k \in \mathbb{Z}})$ .

For example, the outputs of the interpolation operator in (4.3) form a shift-invariant subspace with respect to integer shifts: A shift of  $\hat{x}(t)$  by  $k$  is still in the same subspace. The sampling and interpolation operators we define shortly will also have a shift-invariance property.

**Subspaces of Bandlimited Functions** A special case of shift-invariant subspaces of particular importance in signal processing is the subspace of bandlimited functions; we now define it formally and later we look at forming approximations in these subspaces through sampling and interpolation.

**DEFINITION 4.11 (BANDWIDTH FOR FUNCTIONS)** A function  $x(t) \in \mathcal{L}^2(\mathbb{R})$  is said to have *bandwidth*  $\omega_0$  for the smallest  $\omega_0 \in \mathbb{R}^+$  such that the Fourier transform  $X(\omega)$  satisfies

$$X(\omega) = 0 \quad \text{for all } |\omega| > \frac{\omega_0}{2}. \quad (4.41)$$

**DEFINITION 4.12 (SUBSPACE OF BANDLIMITED FUNCTIONS)** A subspace  $\text{BL}[-\omega_0/2, \omega_0/2] \subset \mathcal{L}^2(\mathbb{R})$  is a *subspace of bandlimited functions* when all  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$  have bandwidth at most  $\omega_0$ .

A subspace of bandlimited functions is shift invariant for any shift  $\tau \in \mathbb{R}^+$ ; in fact, subspaces of bandlimited functions are the only ones that are simultaneously shift invariant for all shifts  $\tau \in \mathbb{R}^+$ . To see shift invariance, take  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$ . Then, (3.56) states that

$$x(t - k\tau) \xrightarrow{\text{FT}} e^{-j\omega k\tau} X(\omega);$$

the Fourier transform is multiplied by a complex exponential, not changing the bandwidth of the shifted function.

#### 4.4.1 Sampling and Interpolation with Orthonormal Functions

In Section 4.1, we introduced sampling and interpolation operators and their combinations, operating on the shift-invariant space of piecewise-constant functions over unit intervals. This was an example where subspaces were spanned with orthonormal functions; we saw that perfect recovery after sampling and interpolation was guaranteed by the specific choice of operators as well as the function subspace. We now formalize this discussion to any sampling interval  $T$  and any  $x(t) \in \mathcal{L}^2(\mathbb{R})$ .

**Sampling** We refer to the operation depicted in Figure 4.3(a), involving filtering with  $\varphi(-t) = g(-t)$  and sampling at  $t = nT$ , as *sampling of the function*  $x(t) \in \mathcal{L}^2(\mathbb{R})$  with *prefilter*  $g(-t)$ , and denote it by  $y_n = (\Phi^*x)_n$ . Through this operation, we move from the larger space  $\mathcal{L}^2(\mathbb{R})$  into the smaller one  $\ell^2(\mathbb{Z})$ .

Generalizing (4.2) for an arbitrary  $T$ , the output of sampling is

$$\begin{aligned}
 y_n &= \langle x(t), g(t - nT) \rangle_t|_{t=nT} = \int_{-\infty}^{\infty} x(t) g(t - nT) dt \\
 &\stackrel{(a)}{=} \int_{-\infty}^{\infty} x(t) \varphi(t - nT) dt = \varphi(-t) * x(t)|_{t=nT} \\
 &= \langle \varphi(t - nT), x(t) \rangle_t = (\Phi^* x)_n,
 \end{aligned} \tag{4.42}$$

where (a) follows from  $\varphi(t) = g(t)$ . Because the domain of the sampling operator  $\Phi^*$  is functions (instead of sequences), we cannot write  $\Phi^*$  as an infinite matrix. However, there is a strong similarity to sequences in that the components of  $\Phi^* x$  are obtained as inner products between  $x(t)$  and shifted versions of a single function  $\varphi(t)$ . We assume  $\{\varphi(t - kT)\}_{k \in \mathbb{Z}}$  to be an orthonormal set,

$$\langle \varphi(t - nT), \varphi(t - kT) \rangle = \delta_{n-k} \quad \Leftrightarrow \quad \Phi^* \Phi = I. \tag{4.43}$$

As before, the sampling operator has a nontrivial null space,  $S^\perp = \mathcal{N}(\Phi^*)$ ; the set  $\{\varphi(t - kT)\}_{k \in \mathbb{Z}}$  spans its orthogonal complement,  $S = \mathcal{N}(\Phi^*)^\perp = \text{span}(\{\varphi(t - kT)\}_{k \in \mathbb{Z}})$ . In other words, when a function  $x(t) \in \mathcal{L}^2(\mathbb{R})$  is sampled, the component that remains is in  $S$  and is captured by  $\Phi^* x$ ; the component that is lost due to sampling is in the null space  $S^\perp$ . Section 4.1 illustrated this sampling operator with  $T = 1$  and  $\varphi(t) = g(t) = \chi_{[0,1)}(t)$ , the indicator function of the unit interval.

**Interpolation** We refer to the operation depicted in Figure 4.3(b), involving point-wise multiplication with a Dirac delta comb function  $s_T(t)$ , (3.7), and filtering with  $g(t)$ , as *interpolation of the sequence*  $y_n \in \ell^2(\mathbb{Z})$  *with postfilter*  $g(t)$ , and denote it by  $\hat{x}(t) = (\Phi y)(t)$ .

Generalizing (4.3) for an arbitrary  $T$ , the output of interpolation is

$$\begin{aligned}
 \hat{x}(t) &= \sum_{n \in \mathbb{Z}} y_n g(t - nT) = \langle y_n, g(t - nT) \rangle_n \\
 &\stackrel{(a)}{=} \langle y_n, \varphi(t - nT) \rangle_n = (\Phi y)(t),
 \end{aligned} \tag{4.44}$$

where (a) follows from  $\varphi(t) = g(t)$ . Denoting the range of  $\Phi$  by  $S$  as before, this subspace is, of course, the same as the orthogonal complement of the null space of the sampling operator, as we have seen earlier. It is also a shift-invariant subspace with respect to integer multiples of  $T$ ; a shift of  $\hat{x}$  by  $nT$  is still in  $S$ . Section 4.1 illustrated this interpolation operator with  $T = 1$  and  $\varphi(t) = g(t) = \chi_{[0,1)}(t)$ .

The way we chose pre- and postfilters, the sampling and interpolation operators are adjoints of each other,

$$\begin{aligned}
 \langle \Phi^* x, y \rangle_{\ell^2} &\stackrel{(a)}{=} \left\langle \int_{-\infty}^{\infty} x(\tau) g(\tau - nT) d\tau, y_n \right\rangle_{\ell^2} \stackrel{(b)}{=} \sum_{n \in \mathbb{Z}} y_n \int_{-\infty}^{\infty} x(\tau) g(\tau - nT) d\tau \\
 &\stackrel{(c)}{=} \int_{-\infty}^{\infty} x(\tau) \sum_{n \in \mathbb{Z}} y_n g(\tau - nT) d\tau \stackrel{(d)}{=} \langle x, \Phi y \rangle_{\mathcal{L}^2},
 \end{aligned} \tag{4.45}$$

where (a) follows from (4.42); (b) from the definition of the  $\ell^2$  inner product; in (c) we interchanged the order of summation and integration; and (d) follows from the definition of the  $\mathcal{L}^2$  inner product as well as (4.44).

**Interpolation Followed by Sampling** Interpolation followed by sampling is described by  $\Phi^*\Phi$  as in Figure 4.4(a),

$$\begin{aligned} (\Phi^*\Phi y)_n &\stackrel{(a)}{=} \Phi^* \sum_{k \in \mathbb{Z}} y_k g(t - kT) \stackrel{(b)}{=} \int_{-\infty}^{\infty} \left( \sum_{k \in \mathbb{Z}} y_k g(t - kT) \right) g(t - nT) dt \\ &\stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} y_k \int_{-\infty}^{\infty} g(t - kT) g(t - nT) dt \stackrel{(d)}{=} y_n, \end{aligned}$$

where (a) follows from the expression for the interpolation operator, (4.44); (b) from the expression for the sampling operator, (4.42); in (c) we interchanged summation and integration; and (d) follows from our assumption, (4.43), that  $\{\varphi(t - kT)\}_{k \in \mathbb{Z}}$  is an orthonormal set. Thus,  $y_n$  is perfectly recovered. Equation (4.43) also shows that the condition for perfect recovery is the same as the set of functions  $\{\varphi(t - kT)\}_{k \in \mathbb{Z}}$  being orthonormal, as in (1.83). This set of functions is not a basis for  $\mathcal{L}^2(\mathbb{R})$ ; instead, it is an orthonormal basis for the subspace  $S$  it spans. Section 4.1 illustrated interpolation followed by sampling with  $T = 1$  and  $\varphi(t) = g(t) = \chi_{[0,1)}(t)$ .

**Sampling Followed by Interpolation** Sampling followed by interpolation is described by  $P = \Phi\Phi^*$  as in Figure 4.4(b).

As for sequences, and given our choice of sampling and interpolation operators,

$$\begin{aligned} P^2 &= \Phi\Phi^*\Phi\Phi^* \stackrel{(a)}{=} \Phi\Phi^* = P, \\ P^* &= (\Phi\Phi^*)^* = \Phi\Phi^* = P, \end{aligned}$$

where (a) follows from (4.43). In other words,  $P$  is idempotent and self-adjoint, that is,  $P$  is an orthogonal projection operator. Then, by Theorem 1.26,  $(Px)(t)$  is the best least-squares approximation of  $x(t)$  in  $S$ ; if  $x(t) \in S$ , sampling followed by interpolation will perfectly recover  $x(t)$ :

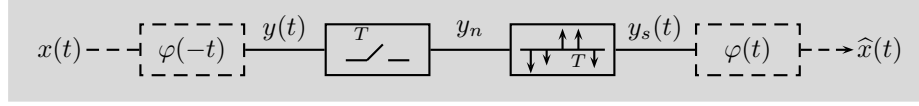
**THEOREM 4.13 (RECOVERY FOR FUNCTIONS)** Given is the system as in Figure 4.4(b) with sampling operator  $\Phi^*$  from (4.42) and interpolation operator  $\Phi$  satisfying (4.43). Then, with  $S = \mathcal{R}(\Phi)$ ,

$$\hat{x}(t) = (\Phi\Phi^*x)(t) \tag{4.46}$$

is the best least-squares approximation of  $x(t)$  in  $S$ , that is,

$$\hat{x}(t) = \min_{x_S(t) \in S} \|x(t) - x_S(t)\|^2, \quad \hat{x}(t) - x(t) \perp S.$$

When  $x(t) \in S$ , then  $\hat{x}(t) = x(t)$ .



**Figure 4.18:** Figure 4.4(b) between sampling prefilter and interpolation postfilter: Sampling of the function  $y(t)$  results in a sampled function  $y_s(t)$ .

Section 4.1 illustrated sampling followed by interpolation with  $T = 1$ ,  $\varphi(t) = g(t) = \chi_{[0,1)}(t)$ , and  $x(t) \in S$ .

#### 4.4.2 Sampling and Interpolation for Bandlimited Functions

Since a subspace of bandlimited functions is also a shift-invariant subspace, everything we have seen thus far holds here as well. In particular, if  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$ , sampling operator  $\Phi^*$  is given by (4.42) and interpolation operator is  $\Phi$ , then by Theorem 4.13,  $x(t)$  is perfectly recovered after sampling followed by interpolation. For  $P$  to orthogonally project onto  $\text{BL}[-\omega_0/2, \omega_0/2]$ , we clearly need sampling and interpolation operators, and thus  $P$ , to be a bit more specific. We now establish what they must be using the machinery from Chapter 3.

We first discuss the sequence of operations between the sampling prefilter and interpolation postfilter in Figure 4.4(b), depicted separately in Figure 4.18: function  $y(t)$  multiplied by the Dirac delta comb function  $s_T(t)$ , (3.7), produces a sampled function  $y_s(t)$ . Using (3.73a),

$$s_T(t) = \sum_{n \in \mathbb{Z}} \delta(t - nT) \xleftrightarrow{\text{FT}} S_T(\omega) = \frac{2\pi}{T} \sum_{k \in \mathbb{Z}} \delta\left(\omega - \frac{2\pi}{T}k\right), \quad (4.47)$$

the sampled function  $y_s(t)$  can be compactly represented as

$$y_s(t) = y(t) s_T(t) = y(t) \sum_{n \in \mathbb{Z}} \delta(t - nT) \stackrel{(a)}{=} \sum_{n \in \mathbb{Z}} y(nT) \delta(t - nT), \quad (4.48)$$

where (a) follows from the sampling property of the Dirac delta function in Table 3.1. Let us now find the Fourier transform of  $y_s(t)$  as well as the DTFT of  $y_n$ :

$$\begin{aligned} y_s(t) &\xleftrightarrow{\text{FT}} Y_s(\omega) = \int_{-\infty}^{\infty} \sum_{n \in \mathbb{Z}} y(nT) \delta(t - nT) e^{-j\omega t} dt \\ &= \sum_{n \in \mathbb{Z}} y(nT) e^{-j\omega nT}, \end{aligned} \quad (4.49a)$$

$$y_n \xleftrightarrow{\text{DTFT}} Y(e^{j\omega}) = \sum_{n \in \mathbb{Z}} y(nT) e^{-j\omega n}. \quad (4.49b)$$

From this, we see that the Fourier transform of the sampled function and the DTFT of the sequence of samples are related by

$$Y(e^{j\omega}) = Y_s\left(\frac{\omega}{T}\right), \quad (4.50)$$



that is, they are the same modulo scaling by  $T$  to make  $Y_s$   $2\pi$ -periodic.

An alternative version of  $Y_s(\omega)$  is often useful:

$$\begin{aligned} Y_s(\omega) &\stackrel{(a)}{=} \frac{1}{2\pi} S_T(\omega) * Y(\omega) \stackrel{(b)}{=} \frac{1}{2\pi} \frac{2\pi}{T} \sum_{k \in \mathbb{Z}} \delta\left(\omega - \frac{2\pi}{T}k\right) * Y(\omega) \\ &\stackrel{(c)}{=} \frac{1}{T} \sum_{k \in \mathbb{Z}} Y\left(\omega - \frac{2\pi}{T}k\right), \end{aligned} \quad (4.51)$$

where (a) follows from the convolution in frequency, (3.65); (b) from (4.47); and (c) from the shifting property of the Dirac delta function (see Table 3.1). This, together with (4.50) leads to the expression connecting the DTFT of a sequence of samples  $y_n$  with the Fourier transform of its underlying continuous-time function  $y(t)$ :

$$Y(e^{j\omega}) = Y_s\left(\frac{\omega}{T}\right) = \frac{1}{T} \sum_{k \in \mathbb{Z}} Y\left(\frac{\omega}{T} - \frac{2\pi}{T}k\right). \quad (4.52)$$

Assume now that  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$  and that the sampling prefilter in Figure 4.4(b) is a simple multiplicative factor of  $\sqrt{T}$ ,<sup>81</sup> so that

$$y(t) = \sqrt{T} x(t), \quad (4.53)$$

leading to

$$Y_s(\omega) \stackrel{(a)}{=} \frac{1}{\sqrt{T}} \sum_{k \in \mathbb{Z}} X\left(\omega - \frac{2\pi}{T}k\right), \quad (4.54)$$

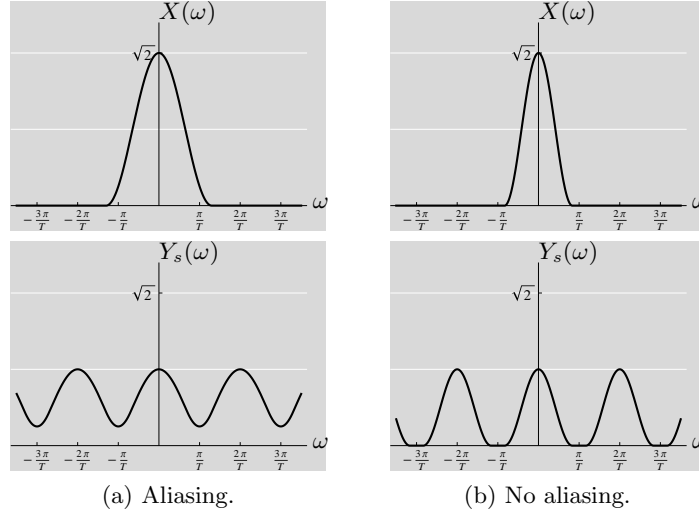
where (a) follows from (4.51). For some LSI postfilter  $g(t)$  to recover  $x(t)$ , a point-wise multiplication of (4.54) by  $G(\omega)$  must yield  $X(\omega)$ . We want the recovery to work for every  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$ , so we cannot count on any property of  $X(\omega)$  other than bandlimitedness (4.41). Thus, multiplication by  $G(\omega)$  must compensate for the  $1/\sqrt{T}$  factor in the  $k = 0$  term of (4.54) and zero out all the other terms.

Whether the multiplication by  $G(\omega)$  will recover  $X(\omega)$  depends on whether the spectral replicas in (4.31) overlap. Figure 4.19 illustrates the two possibilities, which we now discuss in more detail.

**Aliasing** When  $\omega_0 > 2\pi/T$  as in Figure 4.19(a), spectral replicas overlap, and no LSI filtering will succeed in recovering  $x(t)$  for every  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$ . As we have seen, this confusion of frequencies is called aliasing. A graphical example is found in old Western movies, shot at 24 frames/s, where the spokes of wagon wheels seem to be turning backwards due to aliasing (Exercise 4.9).

**EXAMPLE 4.14 (ALIASING OF SINUSOIDS)** Let  $x(t) = \cos(\omega_0 t/2)$ , where  $\omega_0$  is the frequency in radians/s, and let  $\omega_s = 2\pi/T$  be the sampling frequency. In

<sup>81</sup>As for sequences, this means that the sampling prefilter is not present; multiplication by  $\sqrt{T}$  is purely for convenience.



**Figure 4.19:** Sampling at  $t = nT$  followed by multiplication by a Dirac delta comb function of  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$ . (a) When  $\omega_0 > 2\pi/T$ , spectral replicas overlap;  $x(t)$  cannot be recovered by LSI filtering for every  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$ . (b) When  $\omega_0 \leq 2\pi/T$ , spectral replicas do not overlap;  $x(t)$  can be recovered by lowpass filtering by  $g(t)$ . (Illustrated for  $T = 4$ .)

Fourier domain,

$$\cos\left(\frac{\omega_0}{2}t\right) = \frac{e^{j\frac{\omega_0}{2}t} + e^{-j\frac{\omega_0}{2}t}}{2} \xleftrightarrow{\text{FT}} \pi\left(\delta\left(\omega - \frac{\omega_0}{2}\right) + \delta\left(\omega + \frac{\omega_0}{2}\right)\right). \quad (4.55a)$$

Sampling  $x(t)$  with period  $T$ , we get

$$\cos\left(\frac{\omega_0}{2}t\right) \sum_{n \in \mathbb{Z}} \delta(t - nT) \xleftrightarrow{\text{FT}} \pi \sum_{k \in \mathbb{Z}} \delta\left(\omega - k\omega_s - \frac{\omega_0}{2}\right) + \delta\left(\omega - k\omega_s + \frac{\omega_0}{2}\right). \quad (4.55b)$$

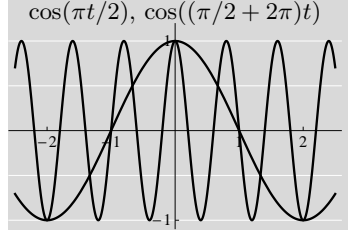
We thus see that  $\cos(\omega_\ell t)$  with  $\omega_\ell = \omega_0/2 + \ell\omega_s$  will have the same Fourier spectrum, or the same samples. Indeed, if  $x(t) = \cos(\omega_0 t/2)$  and  $x'(t) = \cos((\omega_0/2 + \ell\omega_s)t)$ ,

$$x'(nT) = \cos((\omega_0/2 + \ell\omega_s)n(2\pi/\omega_s)) = \cos(\omega_0 nT/2 + \ell 2\pi n) = x(nT);$$

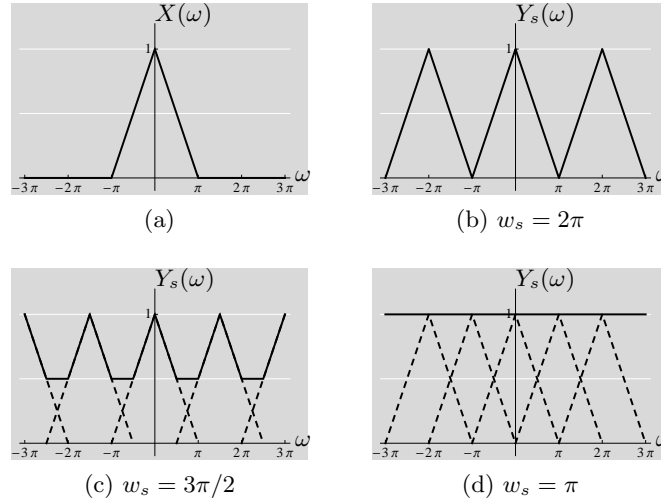
we are not able to tell from which cosine function the samples came (see Figure 4.20 for an example).

**EXAMPLE 4.15 (ALIASED SPECTRA)** Let  $x(t)$  be a function with the Fourier transform as in Figure 4.21(a),

$$X(\omega) = \begin{cases} 1 - |\omega|/\pi, & |\omega| \leq \pi; \\ 0, & \text{otherwise} \end{cases} \xleftrightarrow{\text{FT}} x(t) = \pi \left( \frac{\sin(\frac{\pi}{2}t)}{\frac{\pi}{2}t} \right)^2. \quad (4.56)$$



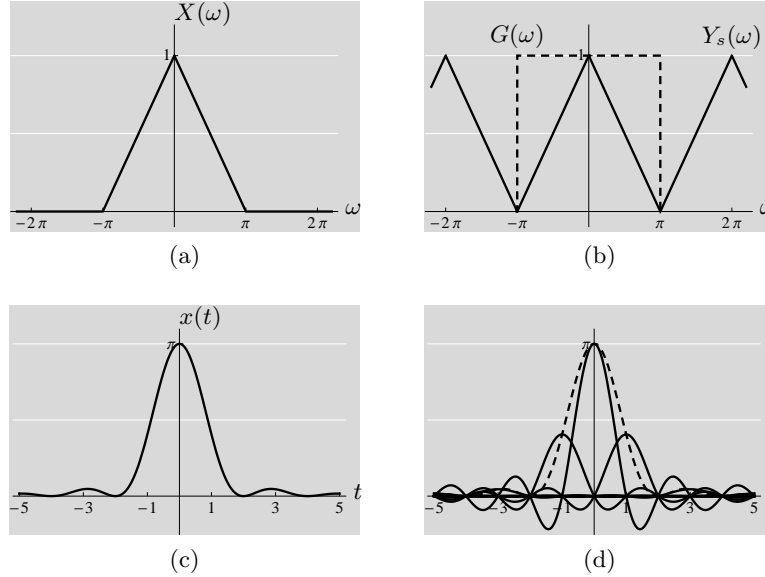
**Figure 4.20:** Illustration of aliasing. Sampling of two cosine functions,  $\cos(\omega_0 t/2)$  and  $\cos((\omega_0/2 + \omega_s)t)$ , with  $\omega_0 = \pi$  and sampling frequency  $\omega_s = 2\pi$ , or  $T = 1$ , produces the same samples.



**Figure 4.21:** A triangle spectrum and various sampled versions. (a) Original spectrum as in (4.56). (b) Spectrum of the sampled function with sampling frequency  $\omega_s = 2\pi$ . (c) Spectrum of the undersampled function with sampling frequency  $\omega_s = 3\pi/2$ . (d) Spectrum of the undersampled function with sampling frequency  $\omega_s = \pi$ .

The same way the hat function is a convolution of two box functions as in Example 3.3, this spectrum can be seen as a convolution of two halfband filters as in Table 3.6, and thus,  $x(t)$  is a multiplication of two sinc functions. Sampling  $x(t)$  at  $\omega_s = 2\pi$ , the triangles stack next to each other with no overlap as in Figure 4.21(b). If we undersample, the triangles will start to overlap, and the original function cannot be recovered from the samples as in Figure 4.21(c) and (d). Take  $\omega_s = \pi$  ( $T = 2$ ). In spectral domain, the triangles sum up to a constant  $1/2$  on  $[0, \pi]$  (from (4.51)) as in Figure 4.21(d). In time domain, this corresponds to a Kronecker delta sequence,  $\pi\delta_n = x(2n)$ , which follows from (4.56).

**Sampling Theorem** When  $\omega_0 \leq 2\pi/T$  as in Figure 4.19(b), spectral replicas do not overlap, and obtaining  $\hat{x} = x$  requires  $G(\omega)$  to be an ideal filter with cut-off



**Figure 4.22:** Illustration of the sampling theorem with  $T = 1$ . (a) The Fourier-domain function  $X(\omega)$ , supported on  $[-\omega_0/2, \omega_0/2] = [-\pi, \pi]$ . (b) Spectrum of the sampled function (solid line), with sampling period  $T = 2\pi/\omega_0 = 1$ , or, equivalently, sampling frequency  $\omega_s = \omega_0 = 2\pi/T = 2\pi$ . Spectral repetitions do not overlap. Interpolated version uses an ideal lowpass filter (dashed line) to extract the base spectrum  $[-\omega_0/2, \omega_0/2] = [-\pi, \pi]$ . (c) The time-domain function  $x(t)$ . (d) The original function  $x(t)$  (dashed line) reconstructed using sinc interpolators (solid lines).

frequency  $\omega_0/2$ . Choosing exactly  $\omega_0 = 2\pi/T$  uniquely determines the postfilter as

$$G(\omega) = \begin{cases} \sqrt{T}, & |\omega| \leq \pi/T; \\ 0, & \text{otherwise,} \end{cases} \quad \xleftrightarrow{\text{FT}} \quad g(t) = \frac{1}{\sqrt{T}} \text{sinc}\left(\frac{\pi}{T}t\right), \quad (4.57)$$

an ideal lowpass filter from Table 2.5. Intuitively, this tells us that  $x(t)$  can be recovered exactly after keeping its values at  $t = nT$  only when its bandwidth is less than  $2\pi/T$ . Since  $\{g(t - kT) = (1/\sqrt{T}) \text{sinc}(\pi(t - kT)/T)\}_{k \in \mathbb{Z}}$  are orthonormal, we can choose the sampling prefilter to be  $g(-t)$ ,<sup>82</sup> leading to an orthogonal projection operator  $P$ . Because  $x(t) \in \text{BL}[-\pi/T, \pi/T]$ , this sampling prefilter has no effect on  $x(t)$ . By construction, the orthonormal set  $\{g(t - kT) = (1/\sqrt{T}) \text{sinc}(\pi(t - kT)/T)\}_{k \in \mathbb{Z}}$  is an orthonormal basis for  $\text{BL}[-\omega_0/2, \omega_0/2]$ .

The frequency  $\omega_s = 2\pi/T$  is called the *sampling frequency*, often also called *Nyquist frequency*; we see that it equals twice the maximum frequency  $\omega_0/2$  of the spectrum of the input function.

This discussion leads us to one of the cornerstone results in signal processing, the sampling theorem,<sup>83</sup> illustrated in Figure 4.22.

<sup>82</sup>We choose a time-reversed version to yield an orthogonal projection operator  $P$ . This time reversal has no effect on the ideal filter since its impulse response is symmetric.

<sup>83</sup>The sampling theorem was formulated and proved by a number of scientists and could bear

**THEOREM 4.14 (SAMPLING THEOREM)** Given is the system as in Figure 4.4(b) with interpolation postfilter  $g(t)$  from (4.57). Then,

$$x(t) = \sum_{n \in \mathbb{Z}} x(nT) \operatorname{sinc}\left(\frac{\pi}{T}(t - nT)\right) \quad \Leftrightarrow \quad x(t) \in \operatorname{BL}\left[-\frac{\pi}{T}, \frac{\pi}{T}\right]. \quad (4.58)$$

*Proof.* Using (4.51), express the reconstructed function in the Fourier domain as

$$\hat{X}(\omega) = G(\omega) \frac{1}{\sqrt{T}} \sum_{k \in \mathbb{Z}} X\left(\omega - \frac{2\pi}{T}k\right).$$

This multiplication simplifies greatly because of the supports of  $G(\omega)$  and  $X(\omega)$ . Since  $x(t) \in \operatorname{BL}[-\pi/T, \pi/T]$ , only the  $k = 0$  term in the summation has a support that overlaps with the  $[-\pi/T, \pi/T]$  support of  $G(\omega)$ . Using the scale factor  $\sqrt{T}$  from (4.57), we get simply  $\hat{X}(\omega) = X(\omega)$ . Note that the sampling prefilter  $g(-t)$  has no effect on the function since it is already bandlimited.

Exercise 4.12 establishes the completeness of  $g(t)$  and its integer shifts as an orthonormal basis for  $\operatorname{BL}[-\pi/T, \pi/T]$ . Then, (4.58) simply expresses  $x(t) \in \operatorname{BL}[-\pi/T, \pi/T]$  as an orthonormal expansion in the orthonormal basis  $(1/\sqrt{T}) \operatorname{sinc}(\pi(t - kT)/T)\}_{k \in \mathbb{Z}}$ .

The sampling theorem gives a sufficient condition for reconstruction of a function from its samples, namely, for a function with bandwidth  $2\pi/T$ , a maximum sampling period of  $T$ , or, a minimum sampling frequency of  $\omega_s = 2\pi/T$ , is needed. This is sometimes not necessary, as shown in the following example.

**EXAMPLE 4.16 (BANDPASS SAMPLING)** Let  $x(t)$  be a function with the Fourier transform as in Figure 4.23(a),

$$X(\omega) = \begin{cases} 3 - |\omega|/\pi, & 2\pi \leq |\omega| \leq 3\pi; \\ 0, & \text{otherwise.} \end{cases} \quad (4.59)$$

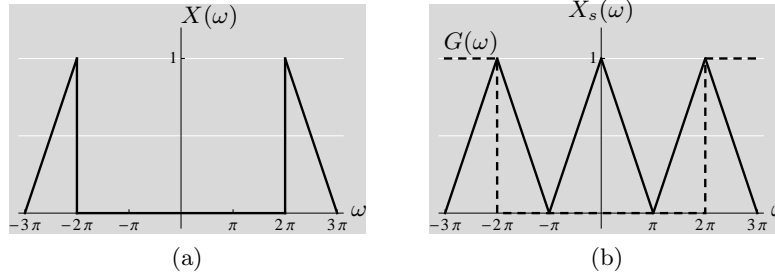
Since the maximum frequency is  $3\pi$ , one might think that a sampling frequency of  $6\pi$  is required. Figure 4.23 shows this not to be true and that a sampling frequency of  $\omega_s = 2\pi$  is sufficient. The spectrum of the sampled version  $X_s(\omega)$ , with  $T = 2\pi/\omega_s = 1$ , is simply

$$X_s(\omega) = \sum_{k \in \mathbb{Z}} X(\omega - 2\pi k).$$

The various parts fill the spectrum without overlapping with each other, creating a triangular spectrum with periodicity  $2\pi$ . For reconstruction, an ideal bandpass filter carves out the correct spectrum on the intervals  $2\pi \leq |\omega| \leq 3\pi$ .

Exercise 4.13 explores bandpass sampling further, showing an orthonormal basis interpretation, a modulation-based solution, and generalizations. The main idea is that for bandpass functions with a total frequency support of  $\omega_0$  on two intervals of size  $\omega_0/2$  each, a sampling frequency of  $\omega_s = \omega_0$  is sufficient.

all their names: Shannon, Kotelnikov, Raabe, Whittaker and Someya, see *Historical Remarks* for more details.



**Figure 4.23:** Bandpass sampling. (a) Original spectrum  $X(\omega)$  in passbands  $2\pi \leq |\omega| \leq 3\pi$ . (b) Spectrum  $X_s(\omega)$  of the sampled function with sampling frequency  $\omega_s = 2\pi$  filtered with an ideal bandpass filter  $G(\omega)$ .

**Bandlimited Approximation of Functions** We now assume that  $x(t) \notin \text{BL}[-\pi/T, \pi/T]$ . Then, as a corollary to Theorem 4.13, since  $P$  is an orthogonal projection operator and  $S = \text{BL}[-\pi/T, \pi/T]$ :

**THEOREM 4.15 (BEST LEAST-SQUARES BANDLIMITED APPROXIMATION)** Given is the system as in Figure 4.4(b) with interpolation postfilter  $g(t)$  from (4.57). Then,

$$\hat{x}(t) = (\Phi\Phi^*x)(t) \quad (4.60)$$

is the best least-squares approximation of  $x(t)$  in  $\text{BL}[-\pi/T, \pi/T]$ , that is,

$$\hat{x}(t) = \min_{x_{\text{BL}}(t) \in \text{BL}[-\pi/T, \pi/T]} \|x(t) - x_{\text{BL}}(t)\|^2, \quad \hat{x}(t) - x(t) \perp \text{BL}[-\pi/T, \pi/T].$$

The effect of this approximation in the Fourier domain is a simple truncation of the spectrum of  $x(t)$  to  $[-\pi/T, \pi/T]$ :

$$\hat{X}(\omega) = \begin{cases} X(\omega), & |\omega| \leq \pi/T; \\ 0, & \text{otherwise.} \end{cases}$$

**Continuous-Time Processing Using Discrete-Time Operators** One more key result around sampling is mathematically simple but has broad technological impact. It shows how to implement continuous-time signal processing operations using discrete-time processing ones. In particular, we show that for convolution.

**PROPOSITION 4.16 (CT CONVOLUTION IMPLEMENTED USING DT PROCESSING)** Let  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$  and  $T = 2\pi/\omega_0$ . The continuous-time convolution

$$y(t) = (g * x)(t),$$

can be computed using the discrete-time convolution

$$\hat{y}_n = (\hat{g} * \hat{x})_n,$$

via

$$y(t) = \sqrt{T} \sum_{n \in \mathbb{Z}} \hat{y}_n \operatorname{sinc}\left(\frac{\pi}{T}(t - nT)\right), \quad (4.61a)$$

with

$$\hat{x}_n = \sqrt{T} x(nT), \quad (4.61b)$$

$$\hat{g}_n = \langle g(t), \operatorname{sinc}\left(\frac{\pi}{T}(t - nT)\right) \rangle, \quad (4.61c)$$

*Proof.* It is easiest to prove this in Fourier domain. We want to show that

$$Y(\omega) = G(\omega)X(\omega)$$

can be obtained as an interpolated discrete-time convolution. Since  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$ , the spectrum of  $G(\omega)$  matters only on this support.

Let us calculate  $\hat{Y}(e^{j\omega})$ ,

$$\begin{aligned} \hat{Y}(e^{j\omega}) &= \hat{G}(e^{j\omega})\hat{X}(e^{j\omega}) \stackrel{(a)}{=} \frac{1}{\sqrt{T}} G\left(\frac{\omega}{T}\right) \sqrt{T} X_s\left(\frac{\omega}{T}\right) \\ &\stackrel{(b)}{=} \frac{1}{T} G\left(\frac{\omega}{T}\right) X\left(\frac{\omega}{T}\right), \quad |\omega| < \pi, \end{aligned} \quad (4.62)$$

where in (a) we observed that (4.61c) is the ideal lowpass filtering of  $g(t)$  to  $[\omega_0/2, \omega_0/2]$  followed by sampling with frequency  $\omega_0$ , and for  $\hat{X}(e^{j\omega})$  we used both (4.50) and (4.61b); and (b) from (4.51) restricted to  $|\omega| < \pi$ .

We can write  $y(t)$  from (4.61a) as

$$y(t) = T \left( \sum_{n \in \mathbb{Z}} \hat{y}_n \delta(t - nT) \right) * \left( \frac{1}{\sqrt{T}} \operatorname{sinc}\left(\frac{\pi}{T}t\right) \right),$$

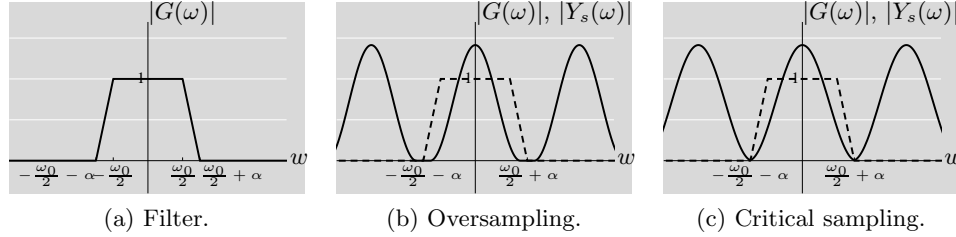
or, in Fourier domain

$$Y(\omega) = \begin{cases} T \hat{Y}(e^{j\omega T}) = G(\omega)X(\omega), & |\omega| < \omega_0/2; \\ 0, & \text{otherwise,} \end{cases}$$

because of both (4.62) as well as the fact that the Fourier transform of  $(1/\sqrt{T}) \operatorname{sinc}(\pi t/T)$  is an ideal lowpass filter as in Table 3.6. Basically, what we did to interpolate  $Y(\omega)$  from  $\hat{Y}(e^{j\omega})$ , was to rescale  $\omega$  so as to get a  $\omega_0$ -periodic function, and then to cut out the base spectrum between  $-\omega_0/2$  and  $\omega_0/2$ .

While we showed the result for convolution, other continuous-time signal processing algorithms having a bandlimited result can also be implemented in discrete time; see Exercise 4.14.

**Approximations to Ideal Filters** In all our developments, we used ideal filters, both as prefilters to obtain perfectly bandlimited functions, or as postfilters, to perfectly interpolate bandlimited functions. However, ideal filters cannot be implemented; moreover, they have slow decay in time domain, of the order  $1/t$ . The solution is to use filters smoother in the frequency domain than the ideal filters. While such filters are more realistic, they will either lead to approximate reconstruction of the input function after sampling and interpolation, or will require oversampling.



**Figure 4.24:** Nonideal, but faster decaying filters and oversampling. (a) Filter  $G(\omega)$  with continuous spectrum obtained from convolving two ideal filters of bandwidth  $\omega_0$  and  $\alpha$ . (b) Input spectrum  $X(\omega)$  with support  $[-\omega_0/2, \omega_0/2]$  and oversampling with sampling frequency  $\omega_s = \omega_0 + \alpha$  to allow perfect interpolation. (c) Input spectrum  $X(\omega)$  with support  $[-\omega_0/2 - \alpha, \omega_0/2 + \alpha]$  and critical sampling with sampling frequency  $\omega_s = \omega_0 + 2\alpha$ , leading to imperfect reconstruction at the boundaries.

**EXAMPLE 4.17 (APPROXIMATIONS TO IDEAL FILTERS)** Consider a filter that is lowpass, but instead of being ideal, it has a spectrum that is continuous, for example,

$$G(\omega) = \begin{cases} 1, & |\omega| \leq \omega_0/2; \\ 1 - (|\omega| - \omega_0/2)/\alpha, & \omega_0/2 \leq |\omega| \leq \omega_0/2 + \alpha; \\ 0, & \text{otherwise,} \end{cases} \quad (4.63a)$$

as in Figure 4.24(a). One way to obtain such a filter is to convolve two ideal filters in frequency, one with cut-off frequency of  $\omega_0/2$  and gain 1, and the other with cut-off frequency of  $\alpha/2$  and gain  $1/\alpha$ , where  $\alpha \ll \omega_0$ . Thus, in time, the impulse response is the product of the two corresponding sinc functions,

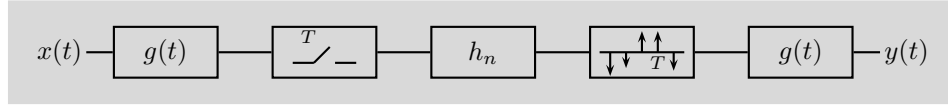
$$g(t) = \frac{\omega_0^2}{4\pi^2\alpha} \operatorname{sinc}\left(\frac{\omega_0}{2}t\right) \operatorname{sinc}\left(\frac{\alpha}{2}t\right). \quad (4.63b)$$

This impulse response decays faster, as  $1/t^2$ , but uses more bandwidth, since the support of  $G(\omega)$  is  $\omega_0 + 2\alpha$ . Then, even if  $X(\omega)$  is bandlimited to  $[-\omega_0/2, \omega_0/2]$ , the sampling and reconstruction using  $G(\omega)$  requires a sampling frequency of  $\omega_s = \omega_0 + \alpha$ , or, oversampling of  $\alpha/\omega_0$ . If  $X(\omega)$  is bandlimited to  $[-\omega_0/2 - \alpha, \omega_0/2 + \alpha]$  and sampled at  $\omega_s = \omega_0 + 2\alpha$ , we lose the tail of the spectrum due to the decay of  $G(\omega)$  at the boundaries. These two cases are shown in Figure 4.24(b) and (c).

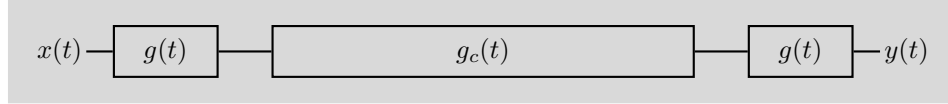
In summary, having a smooth interpolation filter in frequency has a cost, since only the ideal filter allows critical sampling at the minimum sampling frequency together with perfect reconstruction. In practice, most functions of interest have decaying spectra towards their band limit, and thus, the effect of an imperfect reconstruction near the boundary (Figure 4.24(c)) is usually not severe. Solved Exercise 4.4 explores this topic further.

We now describe a practical application of the concepts discussed so far, speech processing in mobile phones.

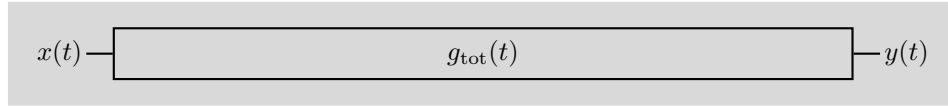




(a) Digital implementation of filtering.



(b) Discrete-time filter in continuous time.



(c) Analog implementation of filtering.

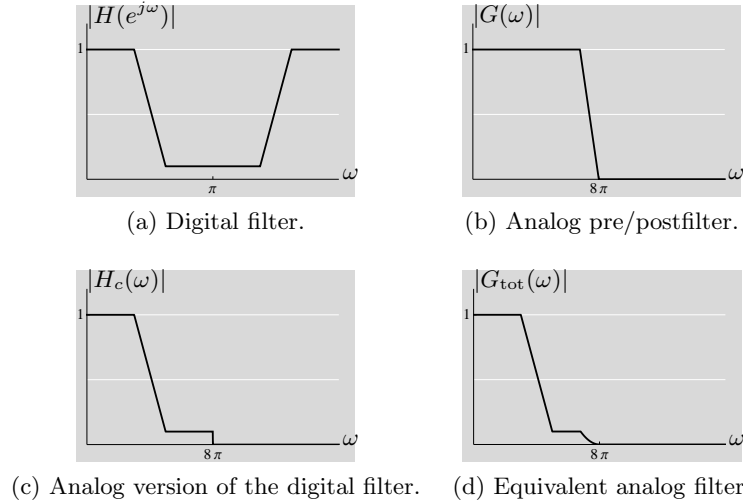
**Figure 4.25:** Analog versus digital implementation of filtering. (a) The input is pre-filtered with  $g(t)$  and sampled with sampling period  $T$ . This sampled version is filtered with a discrete-time lowpass filter  $h_n$ . Finally, the samples are translated to analog, continuous-time domain and postfiltered with  $g(t)$ . (b) Continuous-time filter  $g_c(t)$  is obtained from the discrete-time one  $h_n$ . (c) The input function  $x(t)$  is convolved with  $g_{\text{tot}}(t) = (g * g_c * g)(t)$  to yield the output  $y(t)$ .

**EXAMPLE 4.18 (SPEECH PROCESSING IN MOBILE PHONES)** The bandlimited assumption used in speech and audio processing is based on the fact that humans cannot hear frequencies above 20 kHz. Thus, music for compact disks is sampled at 44 kHz, with a lowpass filter having a passband from  $-20$  kHz to  $20$  kHz and a transition band of  $2$  kHz. For speech, in telephone applications where bandwidth has always been at a premium, a passband from  $0.3$  to  $3.4$  kHz is sufficient for good-quality speech. A sampling frequency of  $f_s = 8$  kHz is used, and the analog pre- and postfilters have a passband of up to  $3.4$  kHz, followed by a smooth transition to very high attenuation between  $3.4$  and  $4$  kHz. For the sake of this example, we assume that these analog filters have a Fourier spectrum like  $G(\omega)$  in (4.63a), with  $\omega_0/2 = 2\pi \cdot 3.4 = 6.8\pi$  krad/s and  $\alpha = 1.2\pi$  krad/s. The sampling frequency is then  $\omega_s = 2\pi f_s = 2\pi \cdot 8 = 16\pi$  krad/s, with  $T = 2\pi/\omega_s = 0.125$  ms sampling period. We now show how filtering in the analog (continuous-time) domain can be implemented using filtering in the digital (discrete-time) domain, as depicted in Figure 4.25(a). For simplicity, in what follows and in the figures, we will express everything in terms of angular frequency in krad/s (that is,  $10^3$  rad/s).

For the input function, assume an exponentially decaying spectrum depicted in Figure 4.27(a),

$$X(\omega) = e^{-2|\omega|/\omega_0}; \quad (4.64)$$

thus  $X(0) = 1$  and  $X(\pm\omega_0/2) = 1/e$ . Before sampling, we prefilter with  $G(\omega)$  from (4.63a), depicted in Figure 4.26(b); the result of this operation is shown in



**Figure 4.26:** Filters involved in analog and digital implementation of filtering in Figure 4.25. The sampling period is  $T = 0.125$  ms, or, the sampling frequency is  $\omega_s = 16\pi$  krad/s; frequency axes are in  $10^3$  rad/s.

Figure 4.27(b). We can now sample the resulting signal yielding a discrete-time signal  $\hat{X}(e^{j\omega})$  depicted in Figure 4.27(c).

We can now filter in the discrete-time domain. Assume we implement a discrete-time lowpass filter that approximates an ideal halfband filter,

$$H(e^{j\omega}) = \begin{cases} 1, & |\omega| \leq 3\pi/8; \\ 10^{-1}, & 5\pi/8 \leq |\omega| < \pi; \end{cases} \quad (4.65)$$

the band  $3\pi/8 \leq |\omega| \leq 5\pi/8$  being a transition band where the filter is unspecified; this filter is depicted in Figure 4.26(a). It corresponds, by frequency scaling (4.50), to a lowpass in continuous time depicted in Figure 4.26(c) with a frequency response

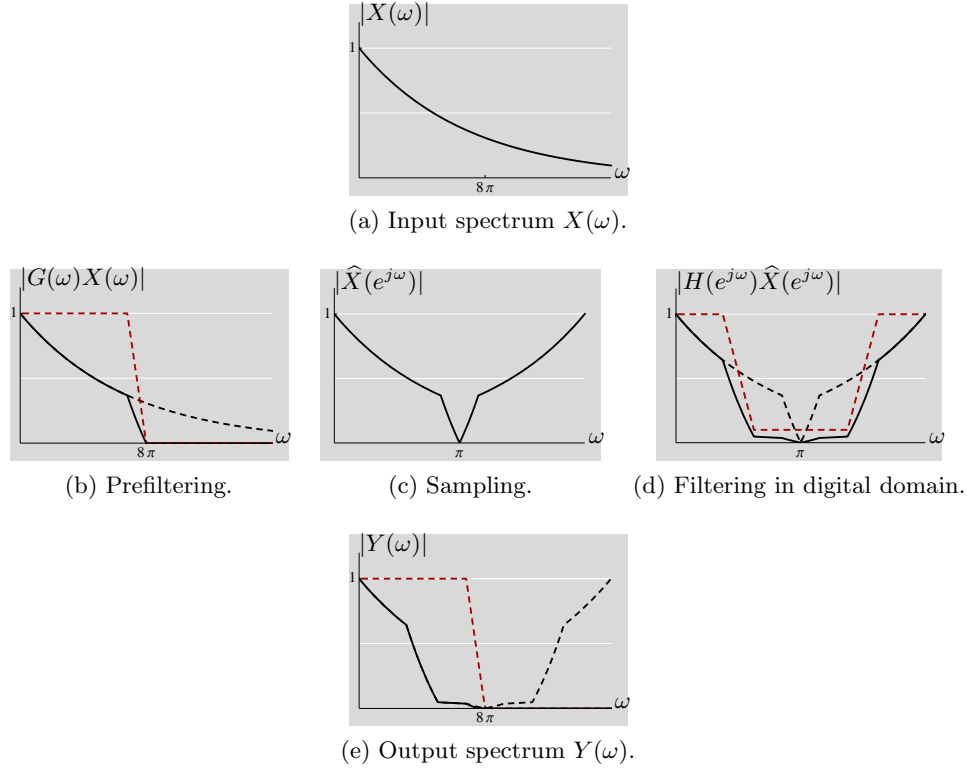
$$H_c(\omega) = H(e^{j\omega T}) = H(e^{j\omega/(8 \cdot 10^3)}) \quad |\omega| \leq 8\pi \cdot 10^3,$$

or a passband up to  $\omega/(8 \cdot 10^3) = 3\pi/8$  krad/s. The discrete-time lowpass filter can be designed using Parks–McClellan optimization; Figure 4.26(a) gives just a conceptual plot.

Finally, the interpolation postfilter  $G(\omega)$  filter cancels the repeated spectra to interpolate the digitally filtered version of the function. Therefore, the overall effect in analog, continuous-time domain, is the product of  $H_c(\omega)$  and the square of  $G(\omega)$ , since it is applied as both a pre- and postfilter,

$$G_{\text{tot}}(\omega) = \begin{cases} H_c(\omega)G^2(\omega), & |\omega| \leq 8\pi \cdot 10^3; \\ 0, & |\omega| > 8\pi \cdot 10^3. \end{cases}$$

The key in this process is the rescaling of the frequency axis, which maps the  $[-\pi, \pi]$  interval of the DTFT to the  $[-\omega_0/2, \omega_0/2]$  interval of the Fourier

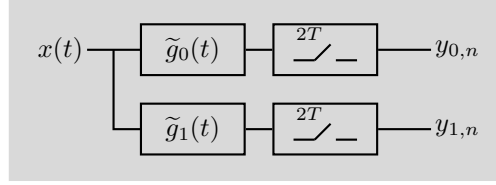


**Figure 4.27:** Analog versus digital implementations of filtering in spectral domain. Filtering the input spectrum  $X(\omega)$  in the analog domain with an analog filter  $G_{\text{tot}}(\omega)$  from Figure 4.26(d) yields the output  $Y(\omega)$ . The same can be achieved by prefiltering with an analog filter  $G(\omega)$  from Figure 4.26(b), sampling, filtering with a digital filter  $H(e^{j\omega})$  from Figure 4.26(a), and interpolating using the postfilter  $G(\omega)$ . Solid black lines are signals at each point in the system from Figure 4.25(a), dashed black lines indicate the spectrum to be filtered while the dashed red lines indicate the corresponding filters. Frequency axes are in kHz.

transform. Note that the scale factor from input to output, while mathematically specified by factors such as  $1/T$  in (4.51) and gains in analog and digital filters, depends on implementation issues such as analog amplifiers and A/D and D/A converter scaling factors.

**Multichannel Sampling** The classic sampling result in Theorem 4.14 has seen many extensions and generalizations, of which we give but a sample. Recall the sampling operator in Figure 4.3(a), but now we have two branches with filtering and sampling yielding a two-channel system as in Figure 4.28. We will call this a two-channel filter bank.

Assume for a moment that  $X(\omega)$  is bandlimited to  $[-\pi, \pi]$ ; this means it can be sampled with sampling frequency  $\omega_s = 2\pi$ , or, sampling period  $T = 1$ .



**Figure 4.28:** Multichannel sampling for  $N = 2$ . The sampling operator consists of two branches with filtering and sampling in parallel, producing two sampled outputs.

However, since we have access to two sampled outputs, it is intuitive that under certain conditions on the filters  $\tilde{G}_0(\omega)$  and  $\tilde{G}_1(\omega)$ , sampling the two channels with a sampling period  $2T$  would be a sufficient representation of  $X(\omega)$ . To see whether this is true, write the spectrum  $Y_i(\omega)$ ,  $i = 0, 1$ , in terms of  $G_i(\omega)$  and  $X(\omega)$ ,

$$\begin{aligned} Y_i(\omega) &\stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} \tilde{G}_i(\omega + k\pi) X(\omega + k\pi) \\ &\stackrel{(b)}{=} \sum_{k \in \mathbb{Z}} \tilde{G}_i(\omega + 2\pi k) X(\omega + 2\pi k) + \sum_{k \in \mathbb{Z}} \tilde{G}_i(\omega + 2\pi k + \pi) X(\omega + 2\pi k + \pi), \end{aligned}$$

where (a) follows from (4.51); and in (b) we split the sum into even-indexed and odd-indexed terms. Since  $Y_i(\omega)$  is periodic with period  $\pi$ , we can consider only one interval  $[0, \pi]$ . Moreover, since  $X(\omega)$  is bandlimited to  $[-\pi, \pi]$ , only two spectral components overlap on  $[0, \pi]$ ,

$$\begin{aligned} Y_0(\omega) &= \tilde{G}_0(\omega) X(\omega) + \tilde{G}_0(\omega - \pi) X(\omega - \pi), \\ Y_1(\omega) &= \tilde{G}_1(\omega) X(\omega) + \tilde{G}_1(\omega - \pi) X(\omega - \pi), \end{aligned}$$

or, in matrix notation, for  $\omega \in [0, \pi]$ ,

$$\begin{bmatrix} Y_0(\omega) \\ Y_1(\omega) \end{bmatrix} = \begin{bmatrix} \tilde{G}_0(\omega) & \tilde{G}_0(\omega - \pi) \\ \tilde{G}_1(\omega) & \tilde{G}_1(\omega - \pi) \end{bmatrix} \begin{bmatrix} X(\omega) \\ X(\omega - \pi) \end{bmatrix} = G(\omega) \begin{bmatrix} X(\omega) \\ X(\omega - \pi) \end{bmatrix}. \quad (4.66)$$

We see that as long as the matrix  $G(\omega)$  is nonsingular on the interval  $[0, \pi]$ , we can recover  $X(\omega)$  on  $[-\pi, \pi]$ . The key is that because  $X(\omega)$  is bandlimited, its under-sampled version by a factor 2 contains only two overlapping copies, and having two versions as in (4.66) allows us to separate them when the matrix  $G(\omega)$  is invertible. We illustrate this in the next two examples.

**EXAMPLE 4.19 (SAMPLING A FUNCTION AND ITS DERIVATIVE)** Let  $x(t) \in \text{BL}[-\pi, \pi]$ , and use a two-channel system as in Figure 4.28 with the identity filter and the derivative filter,

$$\tilde{G}_0(\omega) = 1, \quad \tilde{G}_1(\omega) = j\omega,$$

and sampling at  $t = 2n$ . The spectra of the sampled channel signals are

$$\begin{bmatrix} Y_0(\omega) \\ Y_1(\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ j\omega & j(\omega - \pi) \end{bmatrix} \begin{bmatrix} X(\omega) \\ X(\omega - \pi) \end{bmatrix}.$$

The determinant of the above matrix,  $\det(G(\omega)) = j\pi$ , is a constant, making the system invertible and showing that one can reconstruct a bandlimited function from twice undersampled versions of the function and its derivative.

EXAMPLE 4.20 (SIMPLE NONUNIFORM SAMPLING) Let  $x(t) \in \text{BL}[-\pi, \pi]$ , and use a two-channel system as in Figure 4.28 with the identity filter and the delay of  $\tau \in (0, 2)$  filter,

$$\tilde{G}_0(\omega) = 1, \quad \tilde{G}_1(\omega) = e^{-j\omega\tau},$$

and sampling at  $t = 2n$ . The sampled channel signals are

$$\begin{bmatrix} Y_0(\omega) \\ Y_1(\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ e^{-j\omega\tau} & e^{-j(\omega-\pi)\tau} \end{bmatrix} \begin{bmatrix} X(\omega) \\ X(\omega - \pi) \end{bmatrix}.$$

The determinant of the above matrix is  $\det(G(\omega)) = e^{-j\omega\tau}(e^{j\pi\tau} - 1)$ . This is different from zero for  $\tau \in (0, 2)$ , albeit arbitrarily ill-conditioned as  $\tau$  approaches 0 or 2. This comes as no surprise, since for either no delay  $\tau = 0$ , or delay of  $\tau = 2$ , the samples in the two channels are the same, leading to an undersampled, aliased sampling of the input. As a sanity check, choose  $\tau = 1$ , which should lead to the usual sampling of  $x(t)$ , with even samples in channel 0, and odd ones in channel 1. Then,

$$\begin{bmatrix} Y_0(\omega) \\ Y_1(\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ e^{-j\omega} & -e^{-j\omega} \end{bmatrix} \begin{bmatrix} X(\omega) \\ X(\omega - \pi) \end{bmatrix},$$

and we can recover the input since  $\det(G(\omega)) = -2e^{-j\omega}$ ,

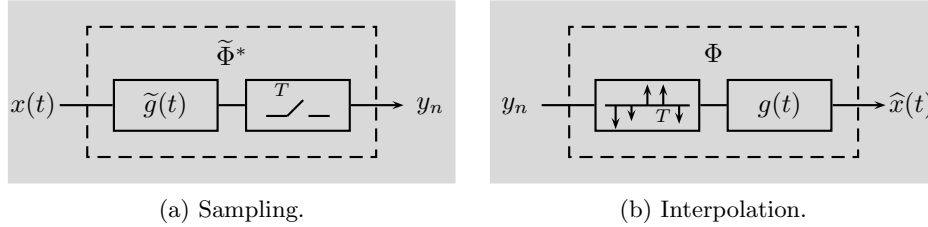
$$\begin{bmatrix} X(\omega) \\ X(\omega - \pi) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & e^{j\omega} \\ 1 & -e^{j\omega} \end{bmatrix} \begin{bmatrix} Y_0(\omega) \\ Y_1(\omega) \end{bmatrix}.$$

The two-channel case in Figure 4.28 just discussed can be readily extended to  $N$ -channels and bandlimited functions of arbitrary bandwidth:

THEOREM 4.17 (MULTICHANNEL SAMPLING (PAPOULIS)) Let  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$ , and let  $T$  be a sampling period with  $T > 2\pi/\omega_0$ . Consider an  $N$ -channel filter bank with filters  $\tilde{g}_i(t)$ ,  $i = 0, 1, \dots, N-1$ , followed by uniform sampling with period  $NT$ . A necessary and sufficient condition for recovery of  $x(t)$  is that the matrix

$$\tilde{G}(\omega) = \begin{bmatrix} \tilde{G}_0(\omega) & \tilde{G}_0(\omega + \frac{2\pi}{NT}) & \dots & \tilde{G}_0(\omega + \frac{2\pi(N-1)}{NT}) \\ \tilde{G}_1(\omega) & \tilde{G}_1(\omega + \frac{2\pi}{NT}) & \dots & \tilde{G}_1(\omega + \frac{2\pi(N-1)}{NT}) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{G}_{N-1}(\omega) & \tilde{G}_{N-1}(\omega + \frac{2\pi}{NT}) & \dots & \tilde{G}_{N-1}(\omega + \frac{2\pi(N-1)}{NT}) \end{bmatrix} \quad (4.67)$$

be nonsingular for  $\omega \in [0, \frac{2\pi}{NT}]$ .



**Figure 4.29:** Sampling and interpolation in  $\mathcal{L}^2(\mathbb{R})$  with nonorthogonal functions.

The proof is a direct extension of what we saw for two channels. Exercise 4.15 explores some ramifications of this result, in particular for derivatives and nonuniform sampling.

#### 4.4.3 Sampling and Interpolation with Nonorthogonal Functions

As for sequences, what we have seen thus far is a classical take on sampling and interpolation; we now expand it to include nonorthogonal functions. These make the geometry more complicated; the sampling and interpolation operators are no longer adjoints of each other, and the appropriate spaces we discussed earlier, the range of the interpolation operator as well as the orthogonal complement of the null space of the sampling operator, are no longer the same.

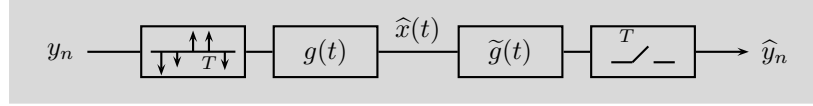
**Sampling** We now refer to the operation depicted in Figure 4.29(a), involving filtering with  $\tilde{g}(t)$  and sampling at  $t = nT$ , as *sampling of the function*  $x(t) \in \mathcal{L}^2(\mathbb{R})$  *with prefilter*  $\tilde{g}(t)$  and denote it by  $y_n = (\tilde{\Phi}^* x)_n$ . Through this operation, we move from the larger space  $\mathcal{L}^2(\mathbb{R})$  into the smaller one  $\ell^2(\mathbb{Z})$ . This time, we do not make an assumption of orthonormality.

Then, the output of sampling is

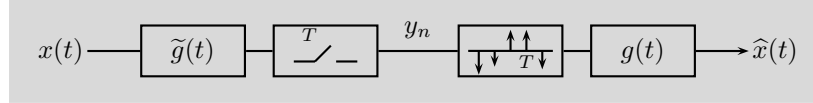
$$\begin{aligned} y_n &= \int_{-\infty}^{\infty} x(t) \tilde{g}(nT - t) dt \stackrel{(a)}{=} \int_{-\infty}^{\infty} x(t) \tilde{\varphi}(t - nT) dt \\ &= \tilde{\varphi}(t) * x(t)|_{t=nT} = \langle \tilde{\varphi}(t - nT), x(t) \rangle_t = (\tilde{\Phi}^* x)_n, \end{aligned} \quad (4.68)$$

where (a) follows from  $\tilde{\varphi}(t) = \tilde{g}(-t)$ . This time, we do not make an assumption of orthonormality. As before, the sampling operator has a nontrivial null space,  $S^\perp = \mathcal{N}(\tilde{\Phi}^*)$ ; the set  $\{\tilde{\varphi}(t - kT)\}_{k \in \mathbb{Z}}$  spans its orthogonal complement,  $\tilde{S} = \mathcal{N}(\tilde{\Phi}^*)^\perp = \text{span}(\{\tilde{\varphi}(t - kT)\}_{k \in \mathbb{Z}})$ . In other words, when a function  $x(t) \in \mathcal{L}^2(\mathbb{R})$  is sampled, the component that remains is in  $\tilde{S}$  and is captured by  $\tilde{\Phi}^* x$ ; the component that is lost due to sampling is in the null space  $\tilde{S}^\perp$ .

**Interpolation** Again, we refer to the operation depicted in Figure 4.3(b), involving pointwise multiplication with a Dirac delta comb function  $s_T(t)$ , (3.7), and filtering with  $g(t)$ , as *interpolation of the sequence*  $y_n \in \ell^2(\mathbb{Z})$  *with postfilter*  $g(t)$ , and denote it by  $\hat{x}(t) = (\Phi y)(t)$ , but this time it is not the adjoint of the sampling operator



(a) Interpolation followed by sampling.



(b) Sampling followed by interpolation.

**Figure 4.30:** Sampling and interpolation in  $\mathcal{L}^2(\mathbb{R})$  with nonorthogonal functions.

$\tilde{\Phi}^*$ . For a given input vector  $y_n \in \ell^2(\mathbb{Z})$ , the interpolation output is a function  $\hat{x}(t) \in \mathcal{L}^2(\mathbb{R})$ ;  $\Phi$  looks the same as in (4.44). When the interpolation operator is specially chosen so that it formally satisfies (4.18), that is, it is the pseudoinverse of  $\tilde{\Phi}$ , then  $\tilde{S} = S$ , by the same arguments as in (4.19).

**EXAMPLE 4.21 (INTERPOLATION IN  $\ell^2(\mathbb{Z})$ )** Choose  $T = 1$  and the postfilter to be the hat function from (3.49a),

$$\varphi(t) = g(t) = \begin{cases} 1 - |t|, & |t| < 1; \\ 0, & \text{otherwise,} \end{cases} \quad (4.69)$$

as the generator of a subspace  $S$  shift-invariant with respect to integer shifts. The subspace  $S = \overline{\text{span}}(\{\varphi(t - k)\}_{k \in \mathbb{Z}})$ , is the space of piecewise linear functions with changes of derivative at the integers.

**Interpolation Followed by Sampling** Interpolation followed by sampling is described by  $\tilde{\Phi}^* \Phi$ , as in Figure 4.30(a). It is possible for  $y_n$  to be perfectly recovered; this happens when

$$\tilde{\Phi}^* \Phi = I \quad \Leftrightarrow \quad \langle \varphi(t - nT), \tilde{\varphi}(t - kT) \rangle = \delta_{n-k}. \quad (4.70)$$

The sampling and interpolation operators are then called *consistent*. Choosing the pseudoinverse in (4.18) for  $\Phi$  would satisfy (4.70); of course, there exist infinitely many other  $\Phi$  we could also use. The above also shows that the condition for perfect recovery is the same as the sets of functions  $\{\varphi(t - kT)\}_{k \in \mathbb{Z}}$  and  $\{\tilde{\varphi}(t - kT)\}_{k \in \mathbb{Z}}$  being biorthogonal, as in (1.102). These sets of functions are not bases for  $\mathcal{L}^2(\mathbb{R})$ ; instead, they form a biorthogonal pair of bases for the subspaces  $S$  and  $\tilde{S}$  they span, respectively.

**EXAMPLE 4.22 (INTERPOLATION FOLLOWED BY SAMPLING IN  $\ell^2(\mathbb{Z})$ )** We start with  $T = 1$ , and assume  $\tilde{g}$  to be as in (4.69). We would like to find a sampling operator such that (4.70) is satisfied.

There exist many choices for  $\tilde{g}(t)$ ; for example, fix the support of  $\tilde{\varphi}(t)$  to be  $[-1/2, 1/2]$  so that (4.70) is satisfied for  $|n - k| > 1$  solely because functions

do not overlap. Next, make  $\tilde{\varphi}(t)$  an even function because, combined with the fact that  $\varphi(t-1)$  is a time-reversed version of  $\varphi(t+1)$ , it will yield the same constraint for  $n-k = \pm 1$ . Many possible parametrization of  $\tilde{\varphi}(t)$  on  $[0, 1/2]$  with two parameters will allow us to determine those parameters uniquely by enforcing (4.70) for  $n-k = 0, 1$ . Assume thus  $\tilde{\varphi}$  to be of the following form:

$$\tilde{\varphi}(t) = \begin{cases} a(b-|t|), & |t| < 1/2; \\ 0, & \text{otherwise,} \end{cases}$$

We thus get the following system of equations:

$$\begin{aligned} \langle \varphi(t), \tilde{\varphi}(t) \rangle &= \int_{-1/2}^{1/2} (1-t) a(b-t) dt = 1, \\ \langle \varphi(t), \tilde{\varphi}(t-1) \rangle &= \int_{1/2}^1 (1-t) a(b-t) dt = 0, \end{aligned}$$

with the solution  $a = 3$ ,  $b = 2/3$ , or

$$\tilde{\varphi}(t) = \begin{cases} 2-3|t|, & |t| < 1/2; \\ 0, & \text{otherwise.} \end{cases} \quad (4.71)$$

Functions  $\varphi(t)$  from (4.69) and  $\tilde{\varphi}(t)$  from (4.71) are plotted in Figure 4.31(a) and (b). With their integer shifts, these functions span different spaces; that is  $\tilde{S} \neq S$ . By inspection of  $\varphi$  and  $\tilde{\varphi}$ , we can see that functions in  $S$  and  $\tilde{S}$  are piecewise linear. The functions in  $S$  are continuous and may have a change of derivative at the integers. The functions in  $\tilde{S}$  are generally discontinuous at the integers and will have changes of derivative at all odd multiples of  $1/2$ . In fact, functions in  $\tilde{S}$  always look like saw blades; Figure 4.31(d) gives an example for  $x(t) = \tilde{\varphi}(t+1) + \tilde{\varphi}(t) + \tilde{\varphi}(t-1)$ .

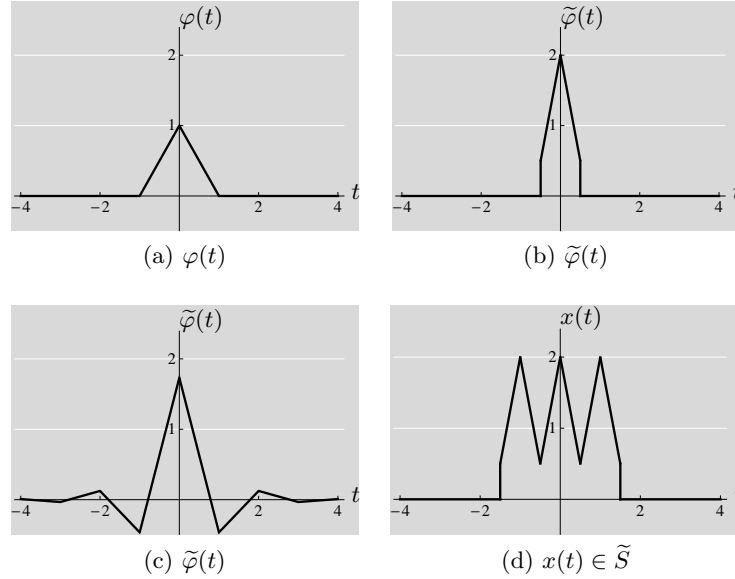
**Sampling Followed by Interpolation** Sampling followed by interpolation is described by  $P = \Phi\Phi^*$ , as in Figure 4.30(b). When the sampling and interpolation operators are consistent as in (4.70), then  $P$  is idempotent, meaning it is a projection operator. It projects onto  $S$ , but the projection is not orthogonal. The approximation error  $x(t) - \hat{x}(t)$  is orthogonal to  $\tilde{S}$  but not to  $S$  (recall Figure 4.11 for a conceptual picture).

Again, for  $P$  to be self-adjoint as well,  $\Phi$  must be chosen to be the pseudoinverse of  $\tilde{\Phi}^*$ , (4.18); the sampling and interpolation operators are then ideally matched, and subspaces  $S$  and  $\tilde{S}$  are identical.

The previous discussion can be summarized as follows:

**THEOREM 4.18 (RECOVERY FOR FUNCTIONS)** Given is the system as in Figure 4.30(b) with sampling operator  $\tilde{\Phi}^*$  from (4.68) and interpolation operator  $\Phi$  from (4.44). Then, with  $S = \mathcal{R}(\Phi)$  and  $\tilde{S} = \mathcal{N}(\tilde{\Phi}^*)^\perp$ :





**Figure 4.31:** Sampling and interpolation for functions. (a) Interpolation postfilter  $\varphi(t)$  from (4.69). (b) Sampling prefilter  $\tilde{\varphi}(t)$  from (4.71) resulting in projection. (c) Sampling prefilter  $\tilde{\varphi}(t)$  from (4.72) with coefficients as in (4.76) resulting in orthogonal projection. (d) A function  $x(t) = \tilde{\varphi}(t+1) + \tilde{\varphi}(t) + \tilde{\varphi}(t-1)$  from  $\tilde{S}$  using  $\tilde{\varphi}(t)$  from (b).

- (i) When  $P = \tilde{\Phi}\tilde{\Phi}^*$  is idempotent, that is, sampling and interpolation operators are consistent, then  $P$  is a projection operator. When  $x(t) \in S$ ,  $x(t)$  is perfectly recovered.
- (ii) When  $P = \tilde{\Phi}\tilde{\Phi}^*$  is idempotent and self-adjoint, that is, sampling and interpolation operators are consistent and ideally matched, then  $P$  is an orthogonal projection operator. When  $x(t) \in S$ ,  $x(t)$  is perfectly recovered.

**EXAMPLE 4.23 (SAMPLING FOLLOWED BY INTERPOLATION IN  $\mathcal{L}^2(\mathbb{R})$ )** For  $P = \tilde{\Phi}\tilde{\Phi}^*$  to be an orthogonal projection operator, apart from consistency, the operators must be ideally matched, that is,  $\tilde{S} = S$ . For this we cannot make arbitrary choices as in Example 4.22; in fact,  $\tilde{\varphi}(t)$  is then uniquely determined.

The key to finding  $\tilde{\varphi}$  such that  $\tilde{S} = S$  is that if  $\tilde{\varphi} \in S$ , then integer shifts of  $\tilde{\varphi}$  generate the same space  $S$ . Thus, let

$$\tilde{\varphi}(t) = \sum_{k \in \mathbb{Z}} \alpha_k \varphi(t - kT) \quad (4.72)$$

for some sequence  $\alpha_k$  to be determined. We do this as follows:

$$\begin{aligned}
 \langle \varphi(t - nT), \tilde{\varphi}(t - kT) \rangle &\stackrel{(a)}{=} \langle \varphi(t - nT), \sum_{\ell \in \mathbb{Z}} \alpha_\ell \varphi(t - kT - \ell T) \rangle \\
 &\stackrel{(b)}{=} \sum_{\ell \in \mathbb{Z}} \alpha_\ell \langle \varphi(t - nT), \varphi(t - kT - \ell T) \rangle \\
 &\stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} \alpha_k a_{n-k-\ell} \stackrel{(d)}{=} \delta_{n-k}, \tag{4.73}
 \end{aligned}$$

where (a) follows from (4.72); (b) from the linearity of the inner product; in (c) we defined an autocorrelation sequence

$$a_m = \langle \varphi(t), \varphi(t + m) \rangle, \tag{4.74}$$

the autocorrelation of  $\varphi(t)$  evaluated at the integers; and (d) follows from our assumption that the operators must be consistent as in (4.70);

To find the sequence  $\alpha_k$ , we recognize (4.73) as a convolution between two sequences  $\alpha_n$  and  $a_n$ . In  $z$ -transform domain, we can rephrase that as

$$\alpha(z)A(z) = 1. \tag{4.75}$$

For  $\varphi(t)$  from (4.69),  $A(z) = (z + 4 + z^{-1})/6$ , and thus

$$\begin{aligned}
 \alpha(z) &= \frac{1}{A(z)} = \frac{6}{z^{-1} + 4 + z} = \frac{6c}{(1 + cz^{-1})(1 + cz)} \\
 &= \frac{6c}{1 - c^2} \left( \frac{1}{1 + cz^{-1}} + \frac{1}{1 + cz} \right),
 \end{aligned}$$

with  $c = 2 - \sqrt{3}$ . This rational  $z$ -transform corresponds to the two-sided sequence

$$\alpha_k = \frac{6c}{1 - c^2} (-c)^{|k|}, \tag{4.76}$$

from which  $\tilde{\varphi}(t)$  follows according to (4.72). Figure 4.31(c) shows this  $\varphi(t)$ .

We will see in the next chapter how this example can be generalized by defining  $S^{(k)}$  as the space of functions that are piecewise polynomial of degree  $k$  with  $k - 1$  continuous derivatives at the integer breakpoints. A biorthogonal basis for  $S^{(k)}$  is given by B-splines of order  $k$ , denoted by  $\beta^{(k)}(t)$ , where  $k$  gives the number of convolutions of the unit box function with itself.

## 4.5 Periodic Functions

In Chapter 3 we studied two classes of functions, those on the real line in Section 3.2.1, and periodic functions in Section 3.2.2. We now consider sampling and interpolation of  $T$ -periodic functions, (3.18), with a square-integrable period and Fourier series coefficients  $X_k$  as in (3.90a). We call the space of such functions  $\mathcal{L}^2([-T/2, T/2])$ ; as we have seen in Chapter 3, operations in such spaces can be

thought of either as operations on periodic functions defined on  $\mathbb{R}$  with a square-integrable period, or operations modulo  $T$  on functions defined on  $\mathcal{L}^2([-T/2, T/2))$ . We will go over a subset of topics we covered for functions on the real line in the last section, those that are of importance in practice.

**Shift-Invariant Subspaces of Periodic Functions** As before, we start by introducing shift-invariant subspaces of  $\mathcal{L}^2([-T/2, T/2))$ .

**DEFINITION 4.19 (SHIFT-INVARIANT SUBSPACES OF  $\mathcal{L}^2([-T/2, T/2))$ )** A subspace  $W \subset \mathcal{L}^2([-T/2, T/2))$  is a *shift-invariant subspace* with respect to shift  $\tau \in [-T/2, T/2)$ , with  $\tau$  integer divisor of  $T$ , when  $x(t) \in W$  implies  $x(t - k\tau) \in W$  for every integer  $k$ . In addition,  $w \in \mathcal{L}^2([-T/2, T/2))$  is called a *generator* of  $W$  when  $W = \text{span}(\{w(t - k\tau)\}_{k \in \mathbb{Z}})$ .

**Subspaces of Bandlimited Periodic Functions** A special case of shift-invariant subspaces of particular importance in signal processing is the subspace of bandlimited periodic functions; we now define it formally and later we look at forming approximations in these subspaces through sampling and interpolation.

**DEFINITION 4.20 (BANDWIDTH FOR PERIODIC FUNCTIONS)** A periodic function  $x(t) \in \mathcal{L}^2([-T/2, T/2))$  is said to have *bandwidth*  $k_0$  for the smallest odd  $k_0 \in \mathbb{Z}^+$  such that the Fourier series  $X_k$  satisfies

$$X_k = 0 \quad \text{for all } |k| > \frac{k_0 - 1}{2}. \quad (4.77)$$

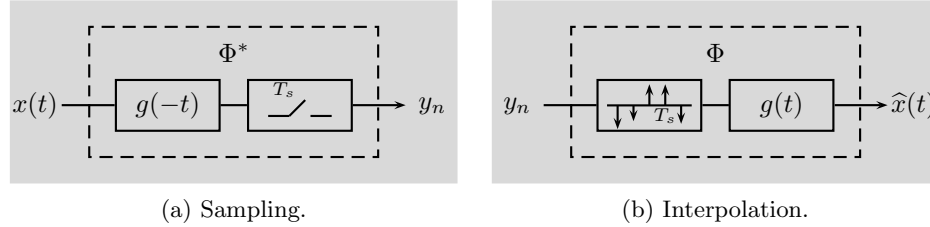
Note that  $k_0$  is odd because for a real function,  $X_k = X_{-k}^*$ , see Table 3.4.

**DEFINITION 4.21 (SUBSPACE OF BANDLIMITED PERIODIC FUNCTIONS)** A subspace  $\text{BL}\{-(k_0 - 1)/2, \dots, (k_0 - 1)/2\} \subset \mathcal{L}^2([-T/2, T/2))$  is a *subspace of bandlimited periodic functions* when all  $x(t) \in \text{BL}\{-(k_0 - 1)/2, \dots, (k_0 - 1)/2\}$  have bandwidth at most  $k_0$ .

A subspace of bandlimited periodic functions is shift invariant for any shift  $\tau \in [-T/2, T/2)$ ; as before, subspaces of bandlimited periodic functions are the only ones that are simultaneously shift invariant for all shifts  $\tau \in [-T/2, T/2)$ . To see shift invariance, take  $x(t) \in \text{BL}\{-(k_0 - 1)/2, \dots, (k_0 - 1)/2\}$ . Then, from (3.102)

$$x(t - n\tau) \xrightarrow{\text{FS}} e^{-j(2\pi/T)kn\tau} X_k;$$

the Fourier transform is multiplied by a complex exponential, not changing the bandwidth of the shifted function.



**Figure 4.32:** Sampling and interpolation in  $\mathcal{L}^2([-T/2, T/2])$  with orthonormal periodic functions.

### 4.5.1 Sampling and Interpolation with Orthonormal Periodic Functions

We now repeat the sequence of steps in examining the operations of sampling and interpolation on periodic functions.

**Sampling** We refer to the operation depicted in Figure 4.32(a), involving circular convolution with  $g(-t)$  and sampling at  $t = nT_s$ , with  $T_s$  integer divisor of  $T$ , as *sampling of the function*  $x(t) \in \mathcal{L}^2([-T/2, T/2])$  with *prefilter*  $g(-t)$ , and denote it by  $y_n = (\Phi^* x)_n$ . Calling the number of sampling points  $k_s = T/T_s = 2k_h + 1$ ,<sup>84</sup> through this operation, we move from the larger space,  $\mathcal{L}^2([-T/2, T/2])$ , into the smaller one,  $\mathbb{C}^{k_s}$ .

The output of sampling is

$$\begin{aligned}
 y_n &= g(-t) \otimes x(t)|_{t=nT_s} = \int_{-T/2}^{T/2} x(t) g(t - nT_s) dt \\
 &\stackrel{(a)}{=} \int_{-T/2}^{T/2} x(t) \varphi(t - nT_s) dt = \varphi(-t) \otimes x(t)|_{t=nT_s} \\
 &= \langle x(t), \varphi(t - nT_s) \rangle_t = (\Phi^* x)_n,
 \end{aligned} \tag{4.78}$$

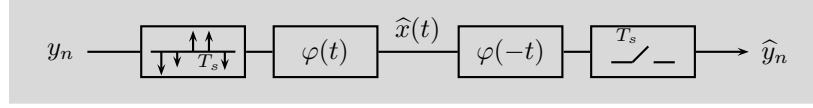
where (a) follows from  $\varphi(t) = g(t)$ . We assume  $\{\varphi(t - nT_s)\}_{n=-k_h}^{k_h}$  to be an orthonormal set,

$$\langle \varphi(t - nT_s), \varphi(t - \ell T_s) \rangle = \delta_{n-\ell} \quad \Leftrightarrow \quad \Phi^* \Phi = I. \tag{4.79}$$

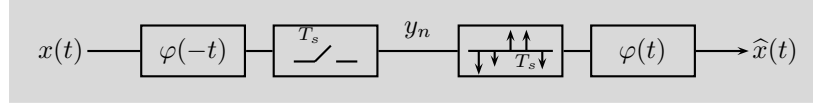
As before, the sampling operator has a nontrivial null space,  $S^\perp = \mathcal{N}(\Phi^*)$ ; the set  $\{\varphi(t - nT_s)\}_{n=-k_h}^{k_h}$  spans its orthogonal complement,  $S = \mathcal{N}(\Phi^*)^\perp = \text{span}(\{\varphi(t - nT_s)\}_{n=-k_h}^{k_h})$ . In other words, when a function  $x(t) \in \mathcal{L}^2([-T/2, T/2])$  is sampled, the component that remains is in  $S$  and is captured by  $\Phi^* x$ ; the component that is lost due to sampling is in the null space  $S^\perp$ .

**Interpolation** We refer to the operation depicted in Figure 4.32(b), involving pointwise multiplication with a Dirac delta comb function  $s_{T_s}(t)$ , (3.7), and cir-

<sup>84</sup>We will assume  $k_s$  to be odd to be consistent with the notion of bandwidth introduced earlier.



(a) Interpolation followed by sampling.



(b) Sampling followed by interpolation.

**Figure 4.33:** Sampling and interpolation in  $\mathcal{L}^2([-T/2, T/2])$  with orthonormal periodic functions.

cular convolution with  $g(t)$ , as *interpolation of the sequence*  $y_n \in \mathbb{C}^{k_s}$  *with postfilter*  $g(t)$ , and denote it by  $\hat{x}(t) = (\Phi y)(t)$ .

The output of interpolation is

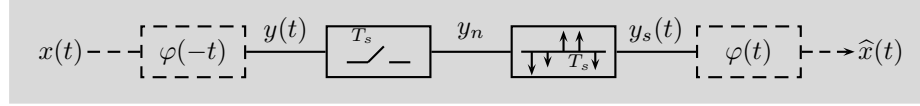
$$\begin{aligned} \hat{x}(t) &= \sum_{n=-k_h}^{k_h} y_n g(t - nT_s) = \langle y_n, g(t - nT_s) \rangle_n \\ &\stackrel{(a)}{=} \langle y_n, \varphi(t - nT_s) \rangle_n = (\Phi y)(t), \end{aligned} \quad (4.80)$$

where (a) follows from  $\varphi(t) = g(t)$ . Denoting the range of  $\Phi$  by  $S$  as before, this subspace is, of course, the same as the orthogonal complement of the null space of the sampling operator, as we have seen earlier. It is also a shift-invariant subspace with respect to integer multiples of  $T_s$ ; a shift of  $\hat{x}$  by  $nT_s$  is still in  $S$ . The way we chose pre- and postfilters, the sampling and interpolation operators are adjoints of each other; the proof mimics (4.45) and is left for Exercise 4.16.

**Interpolation Followed by Sampling** Interpolation followed by sampling is described by  $\Phi^* \Phi$  as in Figure 4.33(a),

$$\begin{aligned} (\Phi^* \Phi y)_n &\stackrel{(a)}{=} \Phi^* \sum_{k=-k_h}^{k_h} y_k g(t - kT_s) \\ &\stackrel{(b)}{=} \int_{-T/2}^{T/2} \left( \sum_{k=-k_h}^{k_h} y_k g(t - kT_s) \right) g(t - nT_s) dt \\ &\stackrel{(c)}{=} \sum_{k=-k_h}^{k_h} y_k \int_{-T/2}^{T/2} g(t - kT_s) g(t - nT_s) dt \stackrel{(d)}{=} y_n, \end{aligned}$$

where (a) follows from the expression for the interpolation operator, (4.80); (b) from the expression for the sampling operator, (4.78); in (c) we interchanged summation and integration; and (d) follows from our assumption, (4.79), that  $\{\varphi(t - kT_s)\}_{k=-k_h}^{k_h}$



**Figure 4.34:** Figure 4.33(b) between sampling prefilter and interpolation postfilter: Sampling of the periodic function  $y(t)$  results in a sampled function  $y_s(t)$ .

is an orthonormal set. Thus,  $y_n$  is perfectly recovered. Equation (4.79) also shows that the condition for perfect recovery is the same as the set of functions  $\{\varphi(t - kT_s)\}_{k=-k_h}^{k_h}$  being orthonormal, as in (1.83). This set of functions is not a basis for  $\mathcal{L}^2([-T/2, T/2])$ ; instead, it is an orthonormal basis for the subspace  $S$  it spans.

**Sampling Followed by Interpolation** Sampling followed by interpolation is described by  $P = \Phi\Phi^*$  as in Figure 4.33(b). As for functions, and given our choice of sampling and interpolation operators,  $P$  is idempotent and self-adjoint, that is,  $P$  is an orthogonal projection operator. Then, by Theorem 1.26,  $(Px)(t)$  is the best least-squares approximation of  $x(t)$  in  $S$ ; if  $x(t) \in S$ , sampling followed by interpolation will perfectly recover  $x(t)$ :

**THEOREM 4.22 (RECOVERY FOR PERIODIC FUNCTIONS)** Given is the system as in Figure 4.33(b) with sampling operator  $\Phi^*$  from (4.78) and interpolation operator  $\Phi$  satisfying (4.79). Then, with  $S = \mathcal{R}(\Phi)$ ,

$$\hat{x}(t) = (\Phi\Phi^*x)(t) \quad (4.81)$$

is the best least-squares approximation of  $x(t)$  in  $S$ , that is,

$$\hat{x}(t) = \min_{x_S(t) \in S} \|x(t) - x_S(t)\|^2, \quad \hat{x}(t) - x(t) \perp S.$$

When  $x(t) \in S$ , then  $\hat{x}(t) = x(t)$ .

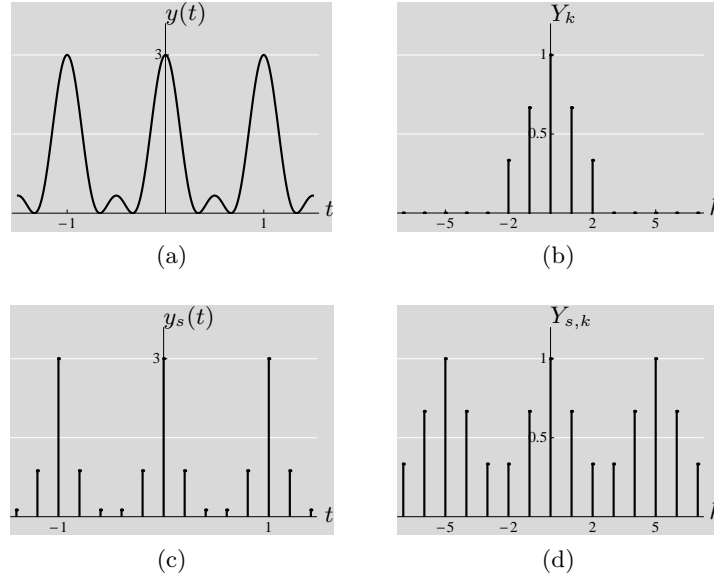
### 4.5.2 Sampling and Interpolation for Bandlimited Periodic Functions

Since a subspace of bandlimited functions is also a shift-invariant subspace, everything we have seen thus far holds here as well. In particular, if  $x(t) \in \text{BL}\{-(k_0 - 1)/2, \dots, (k_0 - 1)/2\}$ , sampling operator  $\Phi^*$  is given by (4.78) and interpolation operator is  $\Phi$ , then by Theorem 4.22,  $x(t)$  is perfectly recovered after sampling followed by interpolation. For  $P$  to orthogonally project onto that bandlimited subspace, we again need the projection operator  $P = \Phi\Phi^*$  to be a bit more specific. We now establish what they must be using the machinery from Chapter 3.

As we did for functions, we first discuss the sequence of operations between the sampling prefilter and interpolation postfilter in Figure 4.33(b), depicted separately

## 4.5. Periodic Functions

431



**Figure 4.35:** Illustration of sampling from Figure 4.34. (a) A bandlimited periodic function  $y(t) \in \text{BL}\{-2, \dots, 2\}$  with period  $T = 1$ . (b) Its bandlimited Fourier series  $Y_k$ . (c) The sampled version of  $y(t)$  with  $T_s = T/5 = 1/5$ , or,  $k_s = 5$ . (d) Its periodic Fourier series  $Y_{s,k}$ .

in Figure 4.34: periodic function  $y(t)$  multiplied by the Dirac delta comb function  $s_{T_s}(t)$ , (3.7), produces a sampled function  $y_s(t)$ . An example function  $y(t)$ , its sampled version  $y_s(t)$  and their respective Fourier series are depicted in Figure 4.35. The Dirac delta comb Fourier series pair is (the proof is left for Exercise 4.17)

$$s_{T_s}(t) = \sum_{n \in \mathbb{Z}} \delta(t - nT_s) = \sum_{\ell \in \mathbb{Z}} \sum_{n=-k_h}^{k_h} \delta(t - \ell T - nT_s), \quad (4.82a)$$

$$S_{T_s,k} = \frac{T^2}{T_s} \sum_{\ell \in \mathbb{Z}} \delta_{k-k_s \ell}, \quad (4.82b)$$

where we represented the periodic function  $s_{T_s}(t)$  as the sum over individual periods. The sampled function  $y_s(t)$  can now be compactly represented as

$$\begin{aligned} y_s(t) &= y(t) s_{T_s}(t) = y(t) \sum_{\ell \in \mathbb{Z}} \sum_{n=-k_h}^{k_h} \delta(t - \ell T - nT_s) \\ &\stackrel{(a)}{=} \sum_{\ell \in \mathbb{Z}} \sum_{n=-k_h}^{k_h} y(\ell T + nT_s) \delta(t - \ell T - nT_s) \\ &\stackrel{(b)}{=} \sum_{\ell \in \mathbb{Z}} \sum_{n=-k_h}^{k_h} y(nT_s) \delta(t - \ell T - nT_s), \end{aligned} \quad (4.83)$$

where (a) follows from the sampling property of the Dirac delta function in Table 3.1; and (b) from the periodicity of  $y(t)$ . Let us now find the Fourier series of  $y_s(t)$  as well as the DFT of  $y_n$ :

$$\begin{aligned}
 y_s(t) &\xleftrightarrow{\text{FS}} Y_{s,k} \stackrel{(a)}{=} \frac{1}{T} \int_{-T/2}^{T/2} \sum_{\ell \in \mathbb{Z}} \sum_{n=-k_h}^{k_h} y(nT_s) \delta(t - \ell T - nT_s) e^{-j(2\pi/T)kt} dt \\
 &\stackrel{(b)}{=} \frac{1}{T} \sum_{n=-k_h}^{k_h} \sum_{\ell \in \mathbb{Z}} \int_{-T/2}^{T/2} y(nT_s) \delta(t - \ell T - nT_s) e^{-j(2\pi/T)kt} dt \\
 &\stackrel{(c)}{=} \frac{1}{T} \sum_{n=-k_h}^{k_h} y(nT_s) \int_{-\infty}^{\infty} \delta(\tau - nT_s) e^{-j(2\pi/T)k\tau} d\tau \\
 &\stackrel{(d)}{=} \frac{1}{T} \sum_{n=-k_h}^{k_h} y(nT_s) e^{-j(2\pi/T)knT_s} \\
 &\stackrel{(e)}{=} \frac{1}{T} \sum_{n=-k_h}^{k_h} y(nT_s) e^{-j(2\pi/k_s)kn}, \tag{4.84a}
 \end{aligned}$$

$$y_n \xleftrightarrow{\text{DFT}} Y_{d,k} \stackrel{(f)}{=} \sum_{n=-k_h}^{k_h} y(nT_s) e^{-j(2\pi/k_s)kn}, \tag{4.84b}$$

where (a) follows from the definition of the Fourier series, (3.90b); in (b) we exchanged the order of summation and integration; in (c) we changed variable  $\tau = t - \ell T$  and concatenated integrals over periods of length  $T$  into a single integral over the real line; (d) follows from the sifting property of the Dirac delta function in Table 3.1; (e) from  $T/T_s = k_s$ ; and (f) from the definition of the DFT, (2.159a). From this, we see that the Fourier series of the sampled function and the DFT of the sequence of samples are related by

$$Y_{d,k} = \frac{1}{T} Y_{s,k}, \tag{4.85}$$

that is, they are the same modulo scaling by  $T$ .

More often, we will use the following alternative version of  $Y_{s,k}$ , as it allows easier analysis of aliasing:

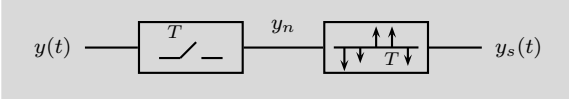
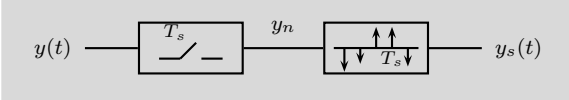
$$Y_{s,k} \stackrel{(a)}{=} S_{T_s,k} * Y_k \stackrel{(b)}{=} \frac{T^2}{T_s} \sum_{\ell \in \mathbb{Z}} \delta_{k-k_s\ell} * Y_k \stackrel{(c)}{=} \frac{T^2}{T_s} \sum_{\ell \in \mathbb{Z}} Y_{k-k_s\ell}, \tag{4.86}$$

where (a) follows from the convolution in frequency, (3.108); (b) from (4.82a); and (c) from the shifting property of the Dirac delta function (see Table 3.1). This expression is the counterpart to (4.51); namely, the Fourier series of the sampled periodic function,  $Y_{s,k}$ , consists of shifted replicas,  $Y_{k-k_s\ell}$  (see Figure 4.35(d)), by integer multiples of  $k_s$ , of the original spectrum  $Y_k$ ; as before, this will lead to a discussion of aliasing, to appear shortly.



## 4.5. Periodic Functions

433

Time domain	Fourier domain
<b>Functions</b>	
	
$y(t)$	$\xleftrightarrow{\text{FT}} Y(\omega)$
$y_n = y(nT)$	$\xleftrightarrow{\text{DTFT}} Y(e^{j\omega}) = \sum_{n \in \mathbb{Z}} y(nT) e^{-j\omega n}$
$y_s(t) = \sum_{n \in \mathbb{Z}} y(nT) \delta(t - nT)$	$\xleftrightarrow{\text{FT}} Y_s(\omega) = \sum_{n \in \mathbb{Z}} y(nT) e^{-j\omega nT}$
	$Y_s(\omega) = \frac{1}{T} \sum_{k \in \mathbb{Z}} Y\left(\omega - \frac{2\pi}{T}k\right)$
	$Y(e^{j\omega}) = Y_s\left(\frac{\omega}{T}\right) = \frac{1}{T} \sum_{k \in \mathbb{Z}} Y\left(\frac{\omega}{T} - \frac{2\pi}{T}k\right)$
$s_T(t) = \sum_{n \in \mathbb{Z}} \delta(t - nT)$	$\xleftrightarrow{\text{FT}} S_T(\omega) = \frac{2\pi}{T} \sum_{k \in \mathbb{Z}} \delta\left(\omega - \frac{2\pi}{T}k\right)$
<b>Periodic functions</b>	
	
$y(t)$	$\xleftrightarrow{\text{FS}} Y_k$
$y_n = y(nT_s)$	$\xleftrightarrow{\text{DFT}} Y_{d,k} = \sum_{n=-k_h}^{k_h} y(nT_s) e^{-j(2\pi/k_s)kn}$
$y_s(t) = \sum_{n \in \mathbb{Z}} y(nT_s) \delta(t - nT_s)$	$\xleftrightarrow{\text{FS}} Y_{s,k} = \frac{1}{T} \sum_{n=-k_h}^{k_h} y(nT_s) e^{-j(2\pi/k_s)kn}$
	$Y_{s,k} = \frac{T^2}{T_s} \sum_{\ell \in \mathbb{Z}} Y_{k-k_s\ell}$
	$Y_{d,k} = \frac{1}{T} Y_{s,k} = \frac{T}{T_s} \sum_{\ell \in \mathbb{Z}} Y_{k-k_s\ell}$
$s_{T_s}(t) = \sum_{\ell \in \mathbb{Z}} \sum_{n=-k_h}^{k_h} \delta(t - \ell T - nT_s)$	$\xleftrightarrow{\text{FS}} S_{T_s,k} = \frac{T^2}{T_s} \sum_{\ell \in \mathbb{Z}} \delta_{k-k_s\ell}$

**Table 4.1:** Summary of sampling relationships. For functions: function  $y(t)$  and its Fourier transform  $Y(\omega)$ , the discrete sampled version  $y_n$  and its DTFT  $Y(e^{j\omega})$ , and the continuous sampled version  $y_s(t)$  and its Fourier transform  $Y_s(\omega)$ . For periodic functions: periodic function  $y(t)$  and its Fourier series  $Y_k$ , the discrete sampled version  $y_n$  and its DFT  $Y_{d,k}$ , and the continuous periodic sampled version  $y_s(t)$  and its Fourier series  $Y_{s,k}$ .

Equation 4.86, together with (4.85) leads to the expression connecting the DFT of a sequence of samples  $y_n$  with the Fourier series of its underlying continuous-time periodic function  $y(t)$ :

$$Y_{d,k} = \frac{1}{T} Y_{s,k} = \frac{T}{T_s} \sum_{\ell \in \mathbb{Z}} Y_{k-k_s\ell}, \quad (4.87)$$

Table 4.1 summarizes these relationships for both functions and periodic functions.

Assume now that  $x(t) \in \text{BL}\{-(k_0 - 1)/2, \dots, (k_0 - 1)/2\}$  and that the sam-

pling prefilter in Figure 4.33(b) is a simple multiplicative factor of  $\sqrt{T_s}/T$ ,<sup>85</sup> so

$$y(t) = \frac{\sqrt{T_s}}{T} x(t), \quad (4.88)$$

leading to

$$Y_{s,k} \stackrel{(a)}{=} \frac{T}{\sqrt{T_s}} \sum_{\ell \in \mathbb{Z}} X_{k-k_s \ell}, \quad (4.89)$$

where (a) follows from (4.86). For some LSI postfilter  $g(t)$  to recover  $x(t)$ , a coefficient-wise multiplication of (4.89) by  $G_k$  must yield  $X_k$ . We want the recovery to work for every  $x(t) \in \text{BL}\{-(k_0 - 1)/2, \dots, (k_0 - 1)/2\}$ , so we cannot count on any property of  $X_k$  other than bandlimitedness (4.77). Thus, multiplication by  $G_k$  must compensate for the  $T/\sqrt{T_s}$  factor in the  $\ell = 0$  term of (4.89) and zero out all the other terms.

Whether the multiplication by  $G_k$  will recover  $X_k$  depends on whether the spectral replicas in (4.86) overlap. We now discuss two possibilities in more detail.

**Aliasing** When  $k_0 > k_s$ , spectral replicas overlap, and no LSI filtering will succeed in recovering  $x(t)$  for every  $x(t) \in \text{BL}\{-(k_0 - 1)/2, \dots, (k_0 - 1)/2\}$ . As we have seen, this confusion of frequencies is called aliasing.

**Sampling Theorem** When  $k_0 \leq k_s$ , spectral replicas do not overlap, and obtaining  $\hat{x} = x$  requires  $G_k$  to be an ideal filter with cut-off frequency  $(k_0 - 1)/2$ . Choosing exactly  $k_0 = k_s = T/T_s$  uniquely determines the postfilter as

$$G_k = \begin{cases} \sqrt{T_s}/T, & |k| \leq k_h; \\ 0, & \text{otherwise,} \end{cases} \quad \xleftrightarrow{\text{FS}} \quad g(t) = \frac{1}{\sqrt{T_s}} \frac{\text{sinc}(\frac{\pi}{T_s}t)}{\text{sinc}(\frac{\pi}{T}t)}. \quad (4.90)$$

The proof that above indeed form a Fourier-series transform pair is left for Solved Exercise 4.5. This function is the *Dirichlet kernel*.

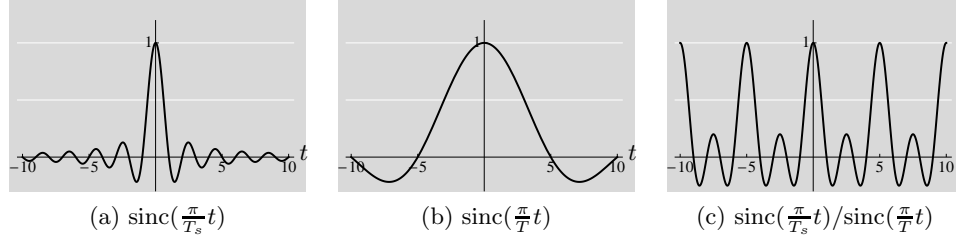
Since  $\{g(t - nT_s)\}_{k=-k_h}^{k_h}$  are orthonormal, we can choose the sampling prefilter to be  $g(-t)$ ,<sup>86</sup> leading to an orthogonal projection operator  $P$ . Because  $x(t) \in \text{BL}\{-k_h, \dots, k_h\}$ , this sampling prefilter has no effect on  $x(t)$ . By construction, the orthonormal set  $\{g(t - nT_s)\}_{k=-k_h}^{k_h}$  is an orthonormal basis for  $\text{BL}\{-k_h, \dots, k_h\}$ .

The frequency  $k_s = T/T_s$  is the Nyquist frequency for periodic functions; we see that it equals twice the maximum frequency  $k_h$  of the spectrum of the input function.

This discussion leads us to the sampling theorem for periodic functions:

<sup>85</sup>As for functions, this means that the sampling prefilter is not present; multiplication by  $\sqrt{T_s}/T$  is purely for convenience.

<sup>86</sup>We choose a time-reversed version to yield an orthogonal projection operator  $P$ . This time reversal has no effect on the ideal filter since its impulse response is symmetric.



**Figure 4.36:** The interpolating function from (4.91) with  $T = 5$  and  $T_s = 1$ .

**THEOREM 4.23 (SAMPLING THEOREM FOR PERIODIC FUNCTIONS)** Given is the system as in Figure 4.33(b) with interpolation postfilter  $g(t)$  from (4.90). Then,

$$x(t) = \sum_{n \in \mathbb{Z}} x(nT_s) \frac{\text{sinc}(\frac{\pi}{T_s}(t - nT_s))}{\text{sinc}(\frac{\pi}{T}(t - nT_s))} \Leftrightarrow x(t) \in \text{BL}\{-k_h, \dots, k_h\}. \quad (4.91)$$

The interpolating Dirichlet kernel does exactly as expected: at a given sampling point  $t = nT_s$ , both numerator and denominator sinc functions equal to 1, while the contribution from the other sinc functions is 0 because the numerator sinc is 1 while the denominator sinc is 1 (see Figure 4.36).

**Bandlimited Approximation of Periodic Functions** We now assume that  $x(t) \notin \text{BL}\{-k_h, \dots, k_h\}$ . Then, as a corollary to Theorem 4.22, since  $P$  is an orthogonal projection operator and  $S = \text{BL}\{-k_h, \dots, k_h\}$ :

**THEOREM 4.24 (BEST LEAST-SQUARES BANDLIMITED APPROXIMATION)** Given is the system as in Figure 4.33(b) with interpolation postfilter  $g(t)$  from (4.90). Then,

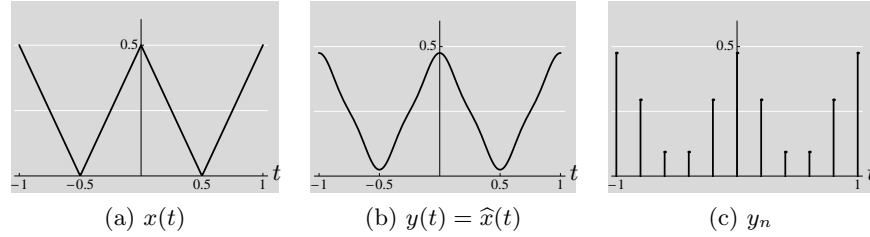
$$\hat{x}(t) = (\Phi\Phi^*x)(t) \quad (4.92)$$

is the best least-squares approximation of  $x(t)$  in  $\text{BL}\{-k_h, \dots, k_h\}$ , that is,

$$\begin{aligned} \hat{x}(t) &= \min_{x_{\text{BL}}(t) \in \text{BL}\{-k_h, \dots, k_h\}} \|x(t) - x_{\text{BL}}(t)\|^2, \\ \hat{x}(t) - x(t) &\perp \text{BL}\{-k_h, \dots, k_h\}. \end{aligned}$$

The effect of this approximation in the Fourier domain is a simple truncation of the spectrum of  $x(t)$  to  $\{-k_h, \dots, k_h\}$ :

$$\hat{X}_k = \begin{cases} X_k, & |k| \leq k_h; \\ 0, & \text{otherwise.} \end{cases} \quad (4.93)$$



**Figure 4.37:** Sampling the triangle wave. (a) The original periodic function  $x(t)$ . (b) Its bandlimited version  $y(t) \in \text{BL}\{-2, \dots, 2\}$ . (c) The sampled version  $y_n$  with  $T_s = 0.2$ . Its interpolated version is the best LS approximation  $\hat{x}(t)$ , and equals  $y(t)$  from (b).

**EXAMPLE 4.24 (SAMPLING THE TRIANGLE WAVE)** Consider  $x(t)$  of period 1 with one period as in (3.114), shown in Figure 4.37(a), with Fourier series coefficients as in (3.115); we repeat both here for completeness

$$x(t) = \left\{ \frac{1}{2} - |t|, \quad |t| \leq \frac{1}{2} \right\} \xleftrightarrow{\text{FS}} X_k = \begin{cases} 1/4, & k = 0; \\ 0, & k \text{ even}, k \neq 0; \\ 1/(\pi k)^2, & k \text{ odd}. \end{cases} \quad (4.94)$$

This function is clearly not bandlimited; by Theorem 4.24, the best least-squares approximation of  $x(t)$  in  $\text{BL}\{-k_h, \dots, k_h\}$  is

$$\begin{aligned} \hat{x}(t) &\stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} \hat{X}_k e^{j2\pi kt} \stackrel{(b)}{=} \sum_{k=-k_h}^{k_h} X_k e^{j2\pi kt} \\ &\stackrel{(c)}{=} \frac{1}{4} + \sum_{m=0}^{2m+1 \leq k_h} \frac{2}{(\pi(2m+1))^2} \cos(2\pi(2m+1)t), \end{aligned} \quad (4.95)$$

where (a) follows from the definition of the inverse Fourier series, (3.90b); (b) from (4.93); and (c) from (4.94) and the summation goes only over odd indices  $k = 2m + 1$ . The bandlimited function  $y(t)$  is illustrated in Figure 4.37(b), while its samples are given in Figure 4.37(c). The best LS approximation in this case is  $\hat{x}(t) = y(t)$ .

If we sampled without lowpass filtering with  $g(-t)$  first, the samples would clearly not be the same as those in Figure 4.37(c).

## 4.6 Stochastic Vectors and Processes

When signals are realizations of stochastic processes, be it in discrete or in continuous time, it is possible to derive sampling theorems similar to the ones we have seen in the deterministic case. We will focus on the bandlimited case, both because of its practical importance and its parallelism to the deterministic setting we saw earlier. In our treatment, we assume WSS processes having a well-defined power spectrum. Furthermore, we require the power spectral density to be continuous, meaning the autocorrelation is in  $\ell^1$  or  $\mathcal{L}^1$ .

### 4.6.1 Finite-Dimensional Stochastic Vectors

Since the sampling and interpolation operators are linear, the effect of additive noise can be analyzed separately from any deterministic vector of interest. We start with vectors and consider the simple case of a white noise vector as in Section 2.8.1; an  $M$ -dimensional random vector  $\mathbf{x} = [x_0 \ x_1 \ \dots \ x_{M-1}]^T$ , whose mean is zero and autocorrelation matrix is  $A_{\mathbf{x}} = E[\mathbf{x}\mathbf{x}^*] = \sigma^2 I$ . How do sampling and interpolation influence the noise characteristics? It is easy to see that the zero-mean property is conserved, what about the correlation properties?

The output of sampling is  $\mathbf{y} = \tilde{\Phi}^* \mathbf{x}$ , and thus

$$A_{\mathbf{y}} = E[\mathbf{y}\mathbf{y}^*] = E[\tilde{\Phi}^* \mathbf{x} \mathbf{x}^* \tilde{\Phi}] = \tilde{\Phi}^* E[\mathbf{x}\mathbf{x}^*] \tilde{\Phi} = \tilde{\Phi}^* A_{\mathbf{x}} \tilde{\Phi} = \sigma^2 \tilde{\Phi}^* \tilde{\Phi}.$$

Thus, the samples of  $\mathbf{y}$  are only uncorrelated when the rows of  $\tilde{\Phi}^*$  are orthogonal; they additionally have equal variance if the rows of  $\tilde{\Phi}^*$  are orthonormal.

The output of interpolation, or  $\hat{\mathbf{x}} = \Phi \mathbf{x}$ , and thus

$$A_{\hat{\mathbf{x}}} = E[\hat{\mathbf{x}}\hat{\mathbf{x}}^*] = E[\Phi \mathbf{x} \mathbf{x}^* \Phi^*] = \Phi E[\mathbf{x}\mathbf{x}^*] \Phi^* = \tilde{\Phi}^* A_{\mathbf{x}} \tilde{\Phi} = \sigma^2 \Phi \Phi^*.$$

Since  $\Phi \Phi^*$  is not of full rank (size  $M \times M$  but rank  $N < M$ ),  $\hat{\mathbf{x}}$  cannot be uncorrelated; no surprise since  $\hat{\mathbf{x}}$  belongs to an  $N$ -dimensional subspace  $S$ .

A similar analysis can be done for the cascade of sampling and interpolation, in either order, showing that perfect reconstruction leads to uncorrelated outputs (see Solved Exercise 4.6).

### 4.6.2 Discrete Bandlimited Stochastic Processes

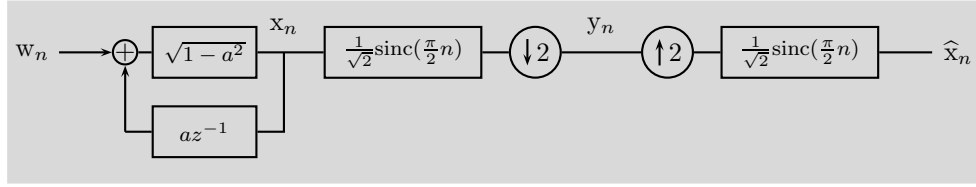
As we said, we assume we are dealing with WSS processes  $x_n$ , that is, those processes whose mean is constant and autocorrelation depends only on the lag  $n$ , as in (2.224). Since we cannot take a DTFT of a discrete stochastic process, as it is neither absolutely, nor square summable, we make assessments based on averages (moments), such as, taking the DTFT of the autocorrelation, or the power spectral density  $A_{\mathbf{x}}(e^{j\omega})$ . We will assume this power spectral density to be continuous here.

To do a sampling and interpolation analysis, we use the results from Section 2.8, such as the power spectral density of the output after downsampling, (2.252), as well as the power spectral density of the output after upsampling, (2.253). We could now follow what we saw for vectors as well as in Section 4.3. We instead concentrate on the sampling theorem for discrete bandlimited stochastic processes, following on Theorem 4.7.

#### THEOREM 4.25 (SAMPLING THEOREM FOR DISCRETE STOCHASTIC PROCESSES)

Given is the system as in Figure 4.14(b) with  $x_n$  a discrete WSS process and interpolation postfilter  $g_n$  from (4.32). Then,

$$x_n = \lim_{L \rightarrow \infty} \sum_{k=-L}^L x_k \operatorname{sinc}\left(\frac{\pi}{N}(n - kN)\right) \Leftrightarrow a_{\mathbf{x},n} \in \operatorname{BL}\left[-\frac{\pi}{N}, \frac{\pi}{N}\right]. \quad (4.96)$$



**Figure 4.38:** An AR-1 system followed by projection onto  $\text{BL}[-\pi/2, \pi/2]$ .

It is also clear that filtering with an ideal lowpass filter with impulse response  $\text{sinc}(\pi n/N)$  and then downsampling by  $N$  before interpolating like in (4.96) computes the best orthogonal projection of the input onto  $\text{BL}[-\pi/N, \pi/N]$ .

**EXAMPLE 4.25 (BANDLIMITING AN AR-1 PROCESS)** Consider the system in Figure 4.38 with an i.i.d. WSS input with variance 1,  $w_n$ . We obtain an AR-1 process  $x_n$  by filtering with a recursive filter as in (2.229). This stochastic process then goes through a sequence of sampling and interpolation with both the prefilter as well as the postfilter being ideal halfband lowpass filters. This multirate system computes the projection of  $x_n$  onto  $\text{BL}[-\pi/2, \pi/2]$ , as we now demonstrate.

The power spectral density of  $x$  is  $A_x(e^{j\omega})$  as computed in (2.234) (see Figure 4.39(a)),

$$A_x(e^{j\omega}) = \frac{1 - a^2}{|1 - ae^{-j\omega}|^2} = \frac{1 - a^2}{1 + a^2 - 2a \cos \omega}, \quad |a| < 1.$$

This power spectral density is then bandlimited to  $[-\pi/2, \pi/2]$  with an ideal halfband filter with gain  $\sqrt{2}$  (see Figure 4.39(b)),

$$\begin{cases} \frac{\sqrt{2}(1-a^2)}{1+a^2-2a \cos \omega}, & |\omega| \leq \pi/2; \\ 0, & \text{otherwise.} \end{cases}$$

This is followed by downsampling by 2. From (2.252) with  $N = 2$ , we get (see Figure 4.39(c)),

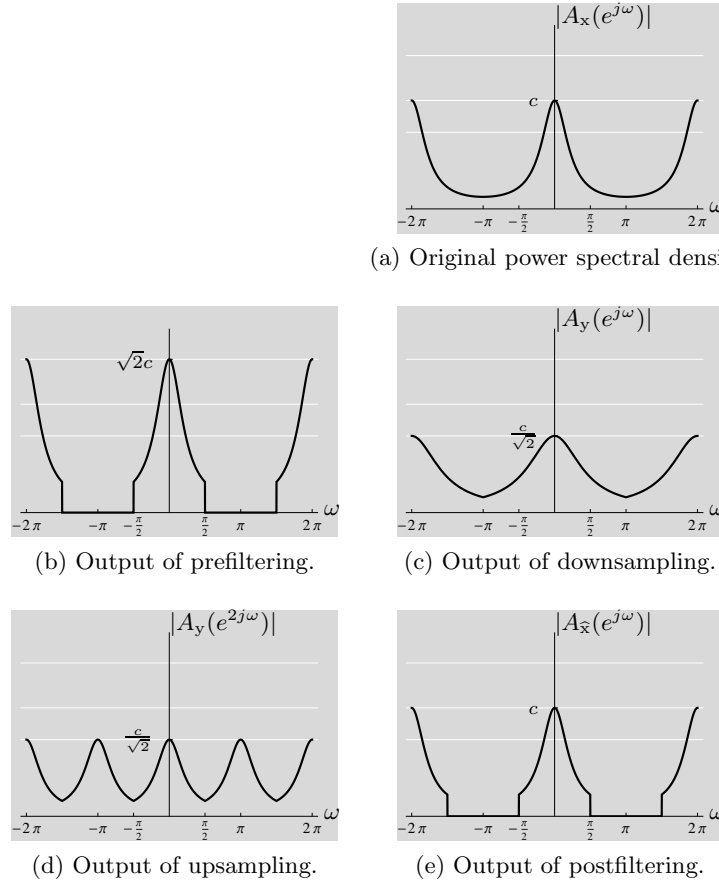
$$A_y(e^{j\omega}) = \frac{1 - a^2}{\sqrt{2}(1 + a^2 - 2a \cos(\omega/2))}.$$

Even though  $\cos(\omega/2)$  is  $4\pi$  periodic,  $A_y(e^{j\omega})$  is  $2\pi$  periodic. After upsampling by 2, from (2.253) we get (see Figure 4.39(d)),

$$\frac{1 - a^2}{\sqrt{2}(1 + a^2 - 2a \cos \omega)},$$

and finally, ideal lowpass filtering produces the bandlimited version of  $x$  (see Figure 4.39(e)),

$$A_{\hat{x}}(e^{j\omega}) = \begin{cases} \frac{1-a^2}{1+a^2-2a \cos \omega}, & |\omega| \leq \pi/2; \\ 0, & \text{otherwise.} \end{cases}$$



**Figure 4.39:** Bandlimiting an AR-1 process as in Figure 4.38. (a) The AR-1 process, followed by (b) prefiltering, (c) downsampling by 2, (d) upsampling by 2, and (e) postfiltering. ( $c = (1 + a)/(1 - a)$ .)

We now verify that  $\hat{x}$  is the best least-squares approximation of  $x$  onto  $\text{BL}[-\pi/2, \pi/2]$ . From the projection theorem, Theorem 1.26, one way of showing this is proving that the approximation error  $x - \hat{x}$  is orthogonal to  $\text{BL}[-\pi/2, \pi/2]$ .

From Table 2.10, we know that if  $x$  is WSS, then the result of filtering followed by downsampling will be WSS as well. Then, that WSS input into upsampling followed by filtering produces a WSS<sub>2</sub> output. We have defined orthogonality for WSS processes in Definition 2.17, and are allowed to compute a power spectral density for WSS processes only; we thus need to prove orthogonality separately for individual polyphase components of  $\hat{x}$ , each of which is WSS.

Take the first polyphase component,  $x_0$ ,

$$E[(x_{2n} - \hat{x}_{2n})\hat{x}_{2n-m}] \stackrel{(a)}{=} c_{x,\hat{x},2n,m} - a_{\hat{x},2n,m} \stackrel{(b)}{=} c_{x,\hat{x}_0,m} - a_{\hat{x}_0,m},$$

where (a) follows from (2.223) and the fact that  $x$  is real; and (b) from  $\hat{x}_0$  being WSS.

TBD: To be finished.

### 4.6.3 Continuous Bandlimited Stochastic Processes

As for discrete stochastic processes, we assume we are dealing with WSS processes  $x(t)$ , that is, those continuous processes whose mean is constant and autocorrelation depends only on the lag  $t$ , as in (3.119). Since we cannot take a Fourier transform of a continuous stochastic process, as it is neither absolutely, nor square integrable, we make assessments based on averages (moments), such as, taking the Fourier transform of the autocorrelation, or the power spectral density  $A_x(\omega)$ . We will assume this power spectral density to be continuous here.

We could now follow what we saw for functions in Section 4.4. We instead concentrate on the sampling theorem for continuous bandlimited stochastic processes, following on Theorem 4.14. The power spectral density of  $y_n = x(nT)$  is given by

$$|A_y(e^{j\omega})|^2 = \frac{1}{T} A_x\left(\frac{\omega}{T}\right), \quad \omega \in [-\pi, \pi]. \quad (4.97)$$

**THEOREM 4.26 (SAMPLING THEOREM FOR CONTINUOUS STOCHASTIC PROCESSES)**  
Given is the system as in Figure 4.4(b) with  $x(t)$  a continuous WSS process and interpolation postfilter  $g(t)$  from (4.57). Then,

$$x(t) = \lim_{L \rightarrow \infty} \sum_{k=-L}^L x(nT) \operatorname{sinc}\left(\frac{\pi}{T}(t - nT)\right) \Leftrightarrow a_x(t) \in \operatorname{BL}\left[-\frac{\pi}{T}, \frac{\pi}{T}\right]. \quad (4.98)$$

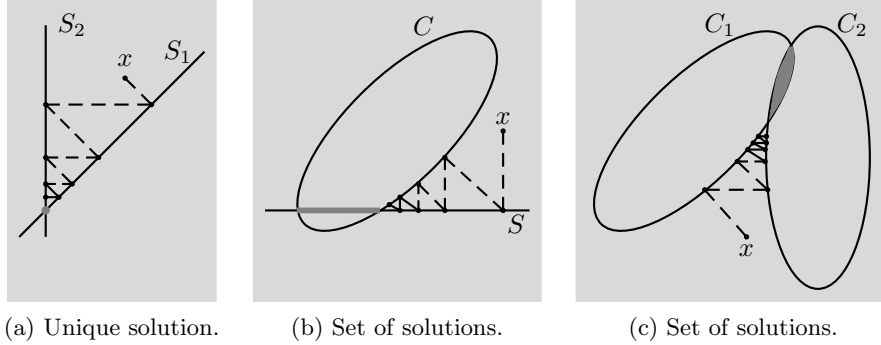
The proof of the theorem is somewhat technical; we omit it here and give pointers in *Further Reading*.

## 4.7 Computational Aspects

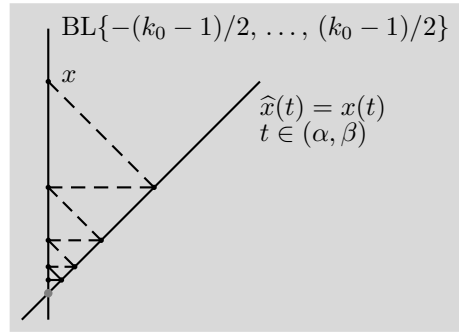
### 4.7.1 Projection Onto Convex Sets

Given  $x$ , the closest point to it on a subspace  $S$  is unique and given by the orthogonal projection  $\hat{x}$ . More generally, given  $x$  in a Hilbert space and a convex subset  $S$ , the closest point to  $x$  on  $S$  is called  $\hat{x}$  and is unique. This fact allows us to find points belonging to the intersection of convex sets by an iterative algorithm called *projection onto convex sets (POCS)*. Intuitively, instead of trying to satisfy all membership constraints at once, one satisfies one constraint at a time;





**Figure 4.40:** Iterative solution of convex constraints using POCS. By iteratively projecting to the closest point, a solution belonging to the intersection is found. (a) The convex sets are subspaces; the solution is unique. (b) Intersection of a general convex set and a subspace; there is a set of possible solutions. (c) Intersection of two general convex sets; there is a set of possible solutions.



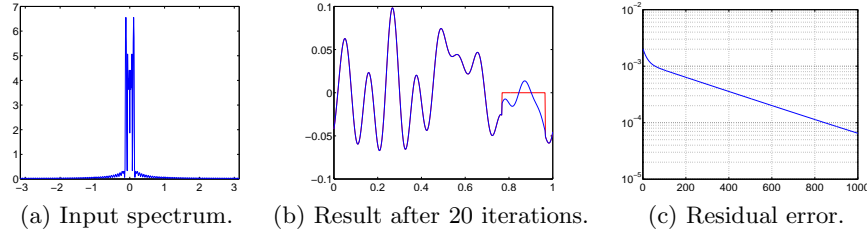
**Figure 4.41:** Illustration of the Papoulis–Gerchberg algorithm on a bandlimited Fourier series of periodic functions with period 1 and partial observation.

because of convexity of the sets, the procedure is guaranteed to converge to one point (not necessarily unique) that belongs to the intersection of all convex sets (see Figure 4.40).

POCS-type algorithms are often used in problems involving bandlimited signals that form a subspace, and have to satisfy some other convex constraint. They are also used because they are simple, and allow to deal with large-size problems. We now discuss a representative example.

**Papoulis–Gerchberg Algorithm** We discuss two instances. We first look into bandlimited Fourier series with partial observation. Let  $x(t) \in \text{BL}\{-(k_0-1)/2, \dots, (k_0-1)/2\}$  be a periodic function of period 1. Let this function be only observed over a subinterval of  $[1/2, 1/2)$ , for example,  $[\alpha, \beta)$ , with  $\alpha, \beta \in (1/2, 1/2)$ .

The Papoulis–Gerchberg algorithm alternates between enforcing two convex constraints:



**Figure 4.42:** Papoulis–Gerchberg algorithm on a bandlimited Fourier series of an example periodic function with period 1 and partial observation.

- (i)  $\hat{x}(t) \in \text{BL}\{-(k_0 - 1)/2, \dots, (k_0 - 1)/2\}$  by truncating the Fourier series as in (4.93); and
- (ii)  $\hat{x}(t) = x(t)$  for  $t \in (\alpha, \beta)$ ; for the missing values, leave  $\hat{x}(t)$  as is.

Figure 4.41) illustrates the procedure. An example of  $x(t)$  with a bandlimited Fourier series where only an interval is actually observed is shown in Figure 4.42. Note the slow convergence at the boundary of the observation interval in Figure 4.42(b).

The same algorithm can be used for images, where a part of the image is missing or corrupted. This is also called the *inpainting* problem.

Let  $x_{n_1, n_2}$  be an  $N \times N$ -size image, and assume its DFT  $X_{k_1, k_2}$  is bandlimited to  $k_{0,1} \times k_{0,2}$  lowpass coefficients. Assume a region of the image is missing, and this region has  $M$  pixels, where  $M < N^2 - k_{0,1}^2 k_{0,2}^2$ . Because the number of missing DFT coefficients  $N^2 - k_{0,1}^2 k_{0,2}^2$  is smaller than the number of measurements  $N^2 - M$ , the solution is uniquely specified (but can be ill-conditioned). The Papoulis–Gerchberg algorithm alternates between enforcing two convex constraints as before:

- (i)  $\hat{x}_{n_1, n_2}$  bandlimited by truncating the DFT; and
- (ii)  $\hat{x}_{n_1, n_2} = x_{n_1, n_2}$  for those  $n_1, n_2$  for which  $x_{n_1, n_2}$  is known; for the missing values, leave  $\hat{x}_{n_1, n_2}$  as is.

Geometrically, the situation is again as in Figure 4.41. We show an example in Figure 4.43. time bandlimited to  $1/2$  of the original bandwidth, is shown in Figure 4.43(a), the same image with missing stripes of size  $1/10N^2$  Figure 4.43(b), and the reconstruction in Figure 4.43(c) (after a few iterations, since ultimately, the reconstruction will be perfect). In this case, the Papoulis–Gerchberg algorithm looks like magic, since it recovers the missing woman.

The Papoulis–Gerchberg algorithm will converge to a unique solution only when specific conditions are satisfied. For example, the number of unknowns (the missing part in the time domain of the signal) should be equal to or less than the number of equations (known frequency part of the signal). Let  $x \in \mathbb{R}^N$  be a discrete-time signal with the DFT  $X \in \mathbb{C}^N$ , and assume that  $x$  is only partially observed.



(a) Original image.



(b) Image bandlimited to half bandwidth.



(c) Image with partial observation.



(d) Reconstructed image.

**Figure 4.43:** Papoulis–Gerchberg algorithm on a bandlimited DFT of an example image with partial observation.

Partition  $x$  and  $X$  as

$$x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}, \quad X = \begin{bmatrix} X_0 \\ X_1 \\ X_2 \end{bmatrix}$$

with  $x_0$  of dimension  $s$ ,  $x_1$  of dimension  $q$  and  $x_2$  of dimension  $N - s - q$ , as well as  $X_0$  of dimension  $N/2 - k$ ,  $X_1$  of dimension  $2k + 1$  and  $X_2$  of dimension  $N/2 - k - 1$ . With the above partitioning, the  $X$  can be written as

$$\begin{bmatrix} X_0 \\ X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} F_{00} & F_{01} & F_{02} \\ F_{10} & F_{11} & F_{12} \\ F_{20} & F_{21} & F_{22} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}.$$

The discrete version of the Papoulis–Gerchberg algorithm solves the following problem: assuming  $x_0$  and  $x_2$  are fixed and  $X_0$  and  $X_2$  are known, find  $X_1$  and  $x_1$ . Note that for the algorithm to converge to a unique solution we need  $q \leq N - 2k - 1$ .

Call the the missing parts at iteration  $n$ ,  $\hat{x}_1^{(n)}$  and  $\hat{X}_1^{(n)}$ . Then next iteration

will produce

$$\begin{aligned}\widehat{X}^{(n+1)} &= \begin{bmatrix} F_{10} & F_{11} & F_{11} \end{bmatrix} \begin{bmatrix} x_0 \\ \widehat{x}_1^{(n)} \\ x_2 \end{bmatrix}, \\ \widehat{x}^{(n)} &= \frac{1}{N} \begin{bmatrix} F_{01}^* & F_{11}^* & F_{21}^* \end{bmatrix} \begin{bmatrix} X_0 \\ \widehat{X}_1^{(n)} \\ X_2 \end{bmatrix}.\end{aligned}$$

Call  $Q = F_{11}F_{11}^*/N$  and  $P = F_{11}^*F_{11}/N$ ; then, the updates are

$$\begin{aligned}\widehat{X}_1^{(n+1)} &= (I - Q)X_1 + Q\widehat{X}_1^{(n)}, \\ \widehat{x}_1^{(n+1)} &= (I - P)x_1 + P\widehat{x}_1^{(n)},\end{aligned}$$

where we have used that  $F_{10}x_0 + F_{12}x_2 = X_1 - F_{11}x_1$  and  $(1/N)(F_{01}^*X_0 + F_{21}^*X_2) = x_1 - (1/N)F_{11}^*X_1$ . With the initial condition  $\widehat{X}_1^{(0)} = \mathbf{0}$ ,

$$\widehat{X}_1^{(n)} = (I - Q^n)X_1. \quad (4.99)$$

From (4.99), we see that the convergence depends on the operator norm of the matrix  $Q$  (its largest eigenvalue). For example, if one chooses  $N = 32$ ,  $s = 7$ ,  $q = 15$  and  $k = 6$ , the largest eigenvalue of  $Q$  is 0.999999998 causing the algorithm to converge slowly.

## Chapter at a Glance

We now summarize the main concepts we have seen in this chapter. They all revolve around sampling, interpolation, and their combinations, between a larger space and a smaller space. In Section 4.2, this larger space is the space of vectors  $\mathbb{C}^M$ , while the smaller space is  $\mathbb{C}^N$ , with  $N < M$ ; in Section 4.3, the larger space is the space of sequences  $\ell^2(\mathbb{Z})$  and the smaller is its subspace; and in Section 4.4, the larger space is the space of functions  $\mathcal{L}^2(\mathbb{R})$  and the smaller is the space of sequences  $\ell^2(\mathbb{Z})$ . By cascading interpolation followed by sampling, or sampling followed by interpolation, we will be able to move from one space to the next and back. It is the match between sampling and interpolation, that will determine the type of recovery possible in each case. These concepts were illustrated in a simple example in Figures 4.6-4.8.

### Sampling and Interpolation with Orthonormal Vectors/Sequences/Functions

- (i) The *sampling* operator  $\Phi^*$  takes an input  $x$  from a larger space and maps it into an output  $y$  in a smaller space. We assume orthonormality, that is,

$$\Phi^* \Phi = I.$$

We call  $S$  call the orthogonal complement of the null space of  $\Phi^*$ ,

$$S = \mathcal{N}(\Phi^*)^\perp.$$

Inputs  $x \in \mathcal{N}(\Phi^*)$  are thus mapped to 0, while inputs  $x \in S$  can be recovered. For inputs  $x \notin S$ , the component in  $\mathcal{N}(\Phi^*)$  is lost due to sampling, while the other component is preserved through  $\Phi^*x$ .

---

#### Sampling $y = \Phi^*x$

input space	input	output	output space
$\mathbb{C}^M$	$x$	$y$	$\mathbb{C}^N \subset \mathbb{C}^M$
$\ell^2(\mathbb{Z})$	$x_n$	$y_n$	$\subset \ell^2(\mathbb{Z})$
$\mathcal{L}^2(\mathbb{R})$	$x(t)$	$y_n$	$\subset \ell^2(\mathbb{Z})$

---

- (ii) The *interpolation* operator  $\Phi$  takes an input  $y$  from a smaller space and maps it into an output  $\hat{x}$  in a larger space. We call  $S$  the range of the interpolation operator  $\Phi$ ,

$$S = \mathcal{R}(\Phi).$$

---

#### Interpolation $\hat{x} = \Phi y$

input space	input	output	output space $S = \mathcal{R}(\Phi)$
$\mathbb{C}^N$	$y$	$\hat{x}$	$S \subset \mathbb{C}^M$
$\ell^2(\mathbb{Z})$	$y_n$	$\hat{x}_n$	$S \subset \ell^2(\mathbb{Z})$
$\ell^2(\mathbb{Z})$	$y_n$	$\hat{x}(t)$	$S \subset \mathcal{L}^2(\mathbb{R})$

---

- (iii) *Interpolation followed by sampling* leads to perfect recovery because of the assumption of orthonormality.

---

#### Interpolation followed by sampling $y = \Phi^* \Phi y$

input	output	reconstruction	property
$y$	$y$	perfect	$\Phi^* \Phi = I$

---

- (iv) *Sampling followed by interpolation* will recover the input perfectly only when  $x \in S$ . Because of the choice of sampling and interpolation operators,  $P = \Phi\Phi^*$  is an orthogonal projection operator.

Sampling followed by interpolation $\hat{x} = \Phi\Phi^* x$			
input	output	reconstruction	property
$x \in S$	$x$	perfect	$\Phi\Phi^* = I$
$x \notin S$	$\hat{x}$	orthogonal projection	

### Sampling and Interpolation with Nonorthogonal Vectors/Sequences/Functions

- (i) The *sampling* operator  $\tilde{\Phi}^*$  takes an input  $x$  from a larger space and maps it into an output  $y$  in a smaller space.  $\tilde{S}$  is the orthogonal complement of the null space of  $\tilde{\Phi}^*$ ,

$$\tilde{S} = \mathcal{N}(\tilde{\Phi}^*)^\perp.$$

Inputs  $x \in \mathcal{N}(\tilde{\Phi}^*)$  are thus mapped to 0, while inputs  $x \in \tilde{S}$  can be recovered. For inputs  $x \notin \tilde{S}$ , the component in  $\mathcal{N}(\tilde{\Phi}^*)$  is lost due to sampling, while the other component is preserved through  $\tilde{\Phi}^*x$ .

- (ii) The *interpolation* operator  $\Phi$  takes an input  $y$  from a smaller space and maps it into an output  $\hat{x}$  in a larger space, as in the orthonormal case. We call  $S$  the range of the interpolation operator  $\Phi$ ,

$$S = \mathcal{R}(\Phi).$$

- (iii) *Interpolation followed by sampling* will not always be identity; when it is, interpolation and sampling are called *consistent*, and the input is perfectly recovered,

Interpolation followed by sampling $\hat{y} = \tilde{\Phi}^*\Phi y$				
input	output	reconstruction	property	
$y$	$y$	perfect	consistent	$\tilde{\Phi}^*\Phi = I$
	$\hat{y}$	not perfect		$\tilde{\Phi}^*\Phi \neq I$

- (iv) *Sampling followed by interpolation* will recover the input perfectly only when  $x \in S$ . When the interpolation operator is the pseudoinverse of the sampling one,

$$\Phi = \tilde{\Phi}(\tilde{\Phi}^*\tilde{\Phi})^{-1},$$

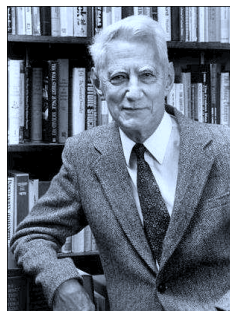
sampling and interpolation are *ideally matched*.<sup>87</sup> If, moreover, they are consistent,  $P = \Phi\tilde{\Phi}^*$  is an orthogonal projection operator. When they are not ideally matched, but consistent only,  $P$  is a projection operator.

Sampling followed by interpolation $\hat{x} = \Phi\tilde{\Phi}^* x$				
input	output	reconstruction	property	
$x \in S$	$x$	perfect	consistent	$\tilde{\Phi}^*\Phi = I$
$x \notin S$	$\hat{x}$	orthogonal projection	ideally matched &	$\Phi = \tilde{\Phi}(\tilde{\Phi}^*\tilde{\Phi})^{-1}$ &
			consistent	$\tilde{\Phi}^*\Phi = I$
	$\hat{x}$	projection	consistent	$\tilde{\Phi}^*\Phi = I$

<sup>87</sup>Throughout,  $\Phi$  and  $\tilde{\Phi}$  could exchange roles;  $\Phi^*$  could be used for sampling,  $\tilde{\Phi}$  for interpolation.

## Historical Remarks

The sampling theorem for bandlimited functions has an interesting history. Long before computers, various functions were tabulated for a set of values in the domain, thus raising the question of interpolation between these points. The use of sinc interpolation for bandlimited functions was first shown by **Edmund T. Whittaker (1873-1956)**, a British mathematician, in 1915. **Harry Nyquist (1889-1976)**, a Swedish electrical engineer, was interested in signaling over bandlimited channels and formulated the celebrated criterion bearing his name in 1928: sampling at twice the maximum frequency uniquely specifies a bandlimited function. In the Russian literature, **Vladimir A. Kotelnikov (1908-2005)**, an information theoretician working in the Soviet Union, proved the sampling theorem independently in 1933 (this is not the only result he and Shannon proved without having been aware of each other). **John M. Whittaker (1905-1984)**, the son of the initial contributor, contributed further results to the interpolation theory on which his father worked. Meanwhile, **Herbert P. Raabe (1909-2004)**, a German electrical engineer, wrote a dissertation in 1939 stating and proving sampling results for bandlimited functions. In 1949, **Someya** in Japan also proved the sampling theorem.



In signal processing and communications, **Claude E. Shannon (1916-2001)**, an American mathematician and engineer, is the one most often connected to the sampling theorem; it often bears his name. In 1948, in his landmark treatise *A Mathematical Theory of Communication*, Shannon formulated the sampling theorem as the first step in digital communications. He did not claim it as its own, however, stating “this is a fact which is common knowledge in the communication art”; Shannon was aware of other formulations, by Whittaker, for example. Apart from the sampling theorem, Shannon is considered the father of information theory, was an accomplished cryptographer, and made numerous contributions to game theory. Shannon’s MS thesis established the design of digital circuits using Boolean algebra, or the foundation of digital design. He

was in contact and collaborated with great mathematicians and engineers of his time; he worked in some of the hallowed institutions of the time, the Institute for Advanced Study in Princeton, NJ, Bell Labs in Murray Hill, NJ, and MIT in Boston, MA.

## Further Reading

Many books and textbooks cover sampling theory and its applications in signal processing and communications, for example, [102]. We also recommend review papers by Jerri [80] and Unser [155]; the latter one, in particular, develops sampling in shift-invariant subspaces.

The basic paper on multichannel sampling as well as the source of Theorem 4.17 is [111]. Nonuniform sampling in its general form is more difficult. Suffice to say that on average, the sampling density has to be at the Nyquist rate, and evenly spread (for example, no accumulation points).

For stationary bandlimited processes, the proof of the sampling theorem goes back to Lloyd [96]; contemporary versions can be found in [175, 113, 88].

For convergence behavior of the Papoulis–Gerchberg algorithm when applied to discrete signals, see [82].

## Exercises with Solutions

### 4.1. Approximation by Piecewise-Constant Functions

Consider the interval  $[0, 1]$  and the linear function  $x(t) = t$ . For any  $k \in \mathbb{N}$ , let  $\hat{x}_k(t)$  be the least-squares approximation of  $x(t)$  among functions that are piecewise constant over intervals  $[n2^{-k}, (n+1)2^{-k})$ ,  $0 \leq n < 2^k$ . Compute  $\|x(t) - \hat{x}_k(t)\|_2^2$  and comment on its behavior as  $k \rightarrow \infty$ .

*Solution:* The set  $\{\varphi_{k,n}(t) = 2^{k/2} \chi_{[n2^{-k}, (n+1)2^{-k})}(t)\}$  forms an orthonormal basis for piecewise constant functions over  $[0, 1]$ . We can then find  $\hat{x}_k(t)$  as

$$\hat{x}_k(t) = \sum_{n=0}^{2^k-1} \langle x, \varphi_{k,n} \rangle \varphi_{k,n}(t) = 2^{-k-1} \sum_{n=0}^{2^k-1} (2n+1) \chi_{[n2^{-k}, (n+1)2^{-k})}(t).$$

So, for example, for  $k=0$ ,  $\hat{x}_0(t) = 1/2$  over  $[0, 1]$ ; for  $k=1$ ,  $\hat{x}_1(t) = 1/4$  over  $[0, 1/2]$  and  $\hat{x}_1(t) = 3/4$  over  $[1/2, 1]$ ; and so on.

Because the basis functions do not overlap, we can compute the error for a single interval and then sum up individual errors:

$$(x(t) - \hat{x}_k(t))^2 \chi_{[n2^{-k}, (n+1)2^{-k})}(t) = \frac{1}{12} 2^{-3k};$$

in other words, for a given  $k$ , the error between  $x(t)$  and  $\hat{x}(t)$  does not depend on  $n$ . Because there are  $2^k$  intervals, the total error is

$$\|x(t) - \hat{x}_k(t)\|_2^2 = 2^k \frac{1}{12} 2^{-3k} = \frac{1}{12} 2^{-2k};$$

as  $k \rightarrow \infty$ , the error will go to 0.

### 4.2. Correcting for Inconsistent Sampling and Interpolation

Given are the sampling operator  $\tilde{\Phi}^*$  and interpolation operator  $\Phi$  in  $\mathbb{C}^M$ , such that  $D = \tilde{\Phi}^* \Phi \neq I$ . We call  $C$  a correction operator when  $P = \Phi C \tilde{\Phi}^*$  is a projection operator. Find  $C$  in terms of  $D$  for it to be a correction operator and note any restrictions on  $D$ .

*Solution:* Our goal is to choose  $C$  so that  $P = \Phi C \tilde{\Phi}^*$  is idempotent,

$$P^2 = (\Phi C \tilde{\Phi}^*)(\Phi C \tilde{\Phi}^*) = \Phi C D C \tilde{\Phi}^*,$$

idempotency is achieved when  $C D C = C$ , that is, when  $C = D^{-1}$ . This correction is possible if and only if  $D$  is invertible.

### 4.3. Null Space of Sampling Operator

Let the sampling  $\tilde{\Phi}^*$  operator be defined as in (4.35), with  $x_n \in \ell^2(\mathbb{Z})$ .

- (i) Find its null space.
- (ii) Show that both  $S$ , the range of the interpolation operator  $\Phi$  from (4.28), as well as  $\tilde{S}$ , the orthogonal complement of the null space of the sampling operator  $\tilde{\Phi}^*$  from (4.35), are shift-invariant subspaces with respect to shift  $N$ .

*Solution:*

- (i) The sampling operator is given by

$$(\tilde{\Phi}^* x)_n = \langle x_k, \tilde{\varphi}_{k-Nn} \rangle_k.$$

Thus, the null space of this operator is given by

$$\mathcal{N}(\tilde{\Phi}^*) = \{x \in \ell^2(\mathbb{Z}) \mid \langle x_k, \tilde{\varphi}_{k-Nn} \rangle_k = 0, n \in \mathbb{Z}\}.$$

- (ii) From (4.28),

$$S = \mathcal{R}(\Phi) = \{\hat{x} \in \ell^2(\mathbb{Z}) \mid \hat{x}_n = \langle y_k, \varphi_{n-Nk} \rangle_k, n \in \mathbb{Z}\}.$$



Clearly, this space is invariant under shift  $N$ , since

$$\hat{x}_{n-N\ell} = \langle y_k, \varphi_{n-Nk-N\ell} \rangle_k$$

still belongs to  $S$ . Similarly, because

$$\tilde{S} = \mathcal{N}(\tilde{\Phi}^*)^\perp = \{y \in \ell^2(\mathbb{Z}) \mid y_n = \sum_{k \in \mathbb{Z}} \alpha_k \tilde{\varphi}_{k-Nn}, n \in \mathbb{Z}\},$$

$\tilde{S}$  is invariant under shift  $N$  since  $y_{n-N\ell}$  still belongs to  $\tilde{S}$ .

#### 4.4. Interpolation of Oversampled Signals

Assume a function  $x(t) \in \text{BL}[-\pi, \pi]$ . If the sampling frequency is chosen at the Nyquist rate,  $\omega_s = 2\pi$ , the interpolation filter is the usual ideal filter, sinc function  $g(t)$  from (4.57) with  $T = 1$ , with slow decay of the order  $1/t$ . If  $x(t)$  is oversampled, then filters with faster decay can be used for interpolating  $x(t)$  from its samples. Such filters are obtained by convolving ideal filters in frequency as in Example 4.17.

- (i) Let  $\omega_s = 3\pi$ . Give the expression for  $g_2(t) = h_1(t)h_2(t)$ , where  $h_i(t)$  are ideal filters with cut-off frequencies  $\omega_1/2$  and  $\omega_2/2$ , respectively. Find what  $\omega_1$  and  $\omega_2$  must be, and verify that  $g_2(t)$  decays as  $1/t^2$ .
- (ii) Let  $\omega_s = 4\pi$ . Give the expression for  $g_3(t) = h_1(t)(h_2(t))^2$ , with same cut-off frequencies as in (i). Find what  $\omega_1$  and  $\omega_2$  must be, and verify that  $g_3(t)$  decays as  $1/t^3$ . Show that  $G_3(\omega)$  has a continuous derivative.
- (iii) Generalize the construction in (ii). Let  $\omega_s = (i+1)\pi$ . Give the expression for  $g_i(t) = h_1(t)(h_2(t))^{(i-1)}$ , with same cut-off frequencies as in (i). Find what  $\omega_1$  and  $\omega_2$  must be, and verify that  $g_i(t)$  decays as  $1/t^i$ . Show that  $G_3(\omega)$  has a continuous  $(i-2)$ th derivative.

*Solution:* Because of  $x(t) \in \text{BL}[-\pi, \pi]$ , and from (4.54), its spectrum sampled at  $\omega_s = k\pi$  will occupy the following set of frequencies:

$$\dots [-(k+1)\pi, -(k-1)\pi] \cup [-\pi, \pi] \cup [(k-1)\pi, (k+1)\pi] \dots \quad (\text{E4.4-1})$$

- (i) From (E4.4-1), we see that with  $\omega_s = 3\pi$ ,  $G_2(\omega)$  must be of the form:

$$G_2(\omega) = \begin{cases} 1, & |\omega| \leq \pi; \\ 0, & |\omega| \geq 2\pi. \end{cases} \quad (\text{E4.4-2})$$

Since  $G_2(\omega) = (H_1 * H_2)(\omega)$ , we can use (E4.4-2) to find the cut-off frequencies  $\omega_1$  and  $\omega_2$ :

$$\frac{\omega_1}{2} + \frac{\omega_2}{2} = 2\pi, \quad \frac{\omega_1}{2} - \frac{\omega_2}{2} = \pi,$$

yielding  $\omega_1 = 3\pi$  and  $\omega_2 = \pi$ , or, from (4.57),

$$g_2(t) = \text{sinc}\left(\frac{3\pi}{2}t\right) \text{sinc}\left(\frac{\pi}{2}t\right);$$

this functions clearly decays as  $1/t^2$ .

- (ii) From (E4.4-1), we see that with  $\omega_s = 4\pi$ ,  $G_3(\omega)$  must be of the form:

$$G_3(\omega) = \begin{cases} 1, & |\omega| \leq \pi; \\ 0, & |\omega| \geq 3\pi. \end{cases} \quad (\text{E4.4-3})$$

Since  $G_3(\omega) = (H_1 * H_2 * H_2)(\omega)$ , we can use (E4.4-3) to find the cut-off frequencies  $\omega_1$  and  $\omega_2$ :

$$\frac{\omega_1}{2} + 2\frac{\omega_2}{2} = 3\pi, \quad \frac{\omega_1}{2} - 2\frac{\omega_2}{2} = \pi,$$

yielding  $\omega_1 = 4\pi$  and  $\omega_2 = \pi$ , or, from (4.57),

$$g_3(t) = \text{sinc}(2\pi t) \left(\text{sinc}\left(\frac{\pi}{2}t\right)\right)^2;$$

this functions clearly decays as  $1/t^3$ .

To show that  $G_3(\omega)$  has a continuous derivative, it is easier to consider it in the time domain first and then transform it back to the frequency domain. From (3.61a),

$$-jtg_3(t) \xleftrightarrow{\text{FT}} \frac{dG_3(\omega)}{d\omega}.$$

The scaling constant  $j$  does not affect continuity so we ignore it,

$$tg_3(t) = \frac{1}{2\pi} \sin(2\pi t) \left( \text{sinc}\left(\frac{\pi}{2}t\right) \right)^2.$$

Thus,  $tg_3(t)$  is a product of two identical sinc functions and a sine function, which corresponds to a Fourier-domain convolution of a triangle window (Fourier-domain pair of the squared sinc function) with a pair of impulses (Fourier-domain pair of the sine function). This is equivalent to adding two triangle windows (which are continuous functions); the derivative is thus continuous. Note that the second derivative is not continuous, since  $t^2g_3(t)$  is a product of a sinc function and two sine functions, which translates into adding four rectangular windows in the frequency domain, and rectangular windows are not continuous.

- (iii) We now generalize what we have seen above. From (E4.4-1), we see that with  $\omega_s = (i+1)\pi$ ,  $G_i(\omega)$  must be of the form:

$$G_i(\omega) = \begin{cases} 1, & |\omega| \leq \pi; \\ 0, & |\omega| \geq i\pi. \end{cases} \quad (\text{E4.4-4})$$

Since  $G_i(\omega) = (H_1 * \underbrace{H_2 * \dots * H_2}_{(i-1) \text{ times}})(\omega)$ , we use (E4.4-4) to find  $\omega_1$  and  $\omega_2$ :

$$\frac{\omega_1}{2} + (i-1)\frac{\omega_2}{2} = i\pi, \quad \frac{\omega_1}{2} - (i-1)\frac{\omega_2}{2} = \pi,$$

yielding  $\omega_1 = (i+1)\pi$  and  $\omega_2 = \pi$ , or, from (4.57),

$$g_i(t) = \text{sinc}\left(\frac{(i+1)\pi}{2}t\right) \left( \text{sinc}\left(\frac{\pi}{2}t\right) \right)^{(i-1)};$$

this functions clearly decays as  $1/t^i$ .

From (3.61a),

$$(-jt)^{(i-2)}g_i(t) \xleftrightarrow{\text{FT}} \frac{d^{(i-2)}G_3(\omega)}{d\omega^{(i-2)}}.$$

Thus,

$$t^{(i-2)}g_i(t) = \frac{1}{i+1} \left( \frac{2}{\pi} \right)^{(i-2)} \sin\left(\frac{(i+1)\pi}{2}t\right) \left( \sin\left(\frac{\pi}{2}t\right) \right)^{(i-3)} \left( \text{sinc}\left(\frac{\pi}{2}t\right) \right)^2.$$

Thus,  $t^{(i-2)}g_i(t)$  is a product of two identical sinc functions and  $(i-2)$  sine functions, which corresponds to a Fourier-domain convolution of a triangle window (Fourier-domain pair of the squared sinc function) with  $(i-2)$  pairs of impulses (Fourier-domain pair of the sine function). This is equivalent to adding triangle windows (which are continuous functions); the  $(i-2)$ th derivative is thus continuous.

#### 4.5. Dirichlet Kernel Proof

Prove that (4.90) is indeed a Fourier-series transform pair.

*Solution:*

$$\begin{aligned}
 g(t) &\stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} G_k e^{j(2\pi/T)kt} \stackrel{(b)}{=} \sum_{k=-k_h}^{k_h} \frac{\sqrt{T_s}}{T} e^{j(2\pi/T)kt} \\
 &\stackrel{(c)}{=} \frac{\sqrt{T_s}}{T} \sum_{n=0}^{k_s-1} e^{j(2\pi/T)(n-k_h)t} \\
 &\stackrel{(d)}{=} \frac{\sqrt{T_s}}{T} e^{-j(2\pi/T)k_h t} \sum_{n=0}^{k_s-1} e^{j(2\pi/T)nt} \\
 &\stackrel{(e)}{=} \frac{\sqrt{T_s}}{T} e^{-j(2\pi/T)k_h t} \frac{1 - e^{j(2\pi/T)k_s t}}{1 - e^{j(2\pi/T)t}} \\
 &\stackrel{(f)}{=} \frac{\sqrt{T_s}}{T} e^{-j(2\pi/T)k_h t} e^{j(\pi/T)k_s t} e^{-j(\pi/T)t} \frac{e^{-j(\pi/T)k_s t} - e^{j(\pi/T)k_s t}}{e^{-j(\pi/T)t} - e^{j(\pi/T)t}} \\
 &\stackrel{(g)}{=} \frac{\sqrt{T_s}}{T} \frac{\sin\left(\frac{\pi}{T}k_s t\right)}{\sin\left(\frac{\pi}{T}t\right)} \stackrel{(f)}{=} \frac{1}{\sqrt{T_s}} \frac{\text{sinc}\left(\frac{\pi}{T}t\right)}{\text{sinc}\left(\frac{\pi}{T}t\right)},
 \end{aligned}$$

where (a) follows from the definition of the Fourier series, (3.90b); (b) from the definition of the ideal filter (4.90); (c) from change of variable  $n = k + k_h$ ; in (d) we just pulled a constant in front of the sum; (e) follows from the finite sum formula (P1.65-1); in (f) we pulled out terms from both numerator and denominator; (g) follows from (2.275); and (f) from the expression for the sinc function, (2.8a), as well as  $k_s = T/T_s$ .

#### 4.6. Effect of Noise on Cascades of Sampling and Interpolation

Let  $\mathbf{x}$  be an  $M$ -dimensional random vector  $\mathbf{x} = [x_0 \ x_1 \ \dots \ x_{M-1}]^T$ , whose mean is zero and autocorrelation matrix is  $A_{\mathbf{x}} = E[\mathbf{x}\mathbf{x}^*] = \sigma^2 I$ . Let also the sampling operator  $\tilde{\Phi}^*$  and interpolation operator  $\Phi$  be consistent. Find the mean and the autocorrelation matrix of the outputs of the cascades

- (i) interpolation followed by sampling,  $\mathbf{y} = \tilde{\Phi}^* \Phi \mathbf{x}$ ; and
- (ii) sampling followed by interpolation,  $\hat{\mathbf{x}} = \Phi \tilde{\Phi}^* \mathbf{x}$ .

*Solution:*

- (i) The mean of the output of interpolation followed by sampling is

$$E[\mathbf{y}] = E[\tilde{\Phi}^* \Phi \mathbf{x}] = E[\mathbf{x}] = 0,$$

because  $\tilde{\Phi}^*$  and  $\Phi$  are consistent ( $\tilde{\Phi}^* \Phi = I$  from (4.38)). Similarly, the autocorrelation matrix is

$$E[\mathbf{y}\mathbf{y}^*] = E[\mathbf{x}\mathbf{x}^*] = \sigma^2 I.$$

- (ii) The mean of the output of sampling followed by interpolation is

$$E[\hat{\mathbf{x}}] = E[\Phi \tilde{\Phi}^* \mathbf{x}] = \Phi \tilde{\Phi}^* E[\mathbf{x}] = 0.$$

The autocorrelation matrix is

$$E[\hat{\mathbf{x}}\hat{\mathbf{x}}^*] = E[\Phi \tilde{\Phi}^* \mathbf{x} \mathbf{x}^* \tilde{\Phi} \Phi^*] = \Phi \tilde{\Phi}^* E[\mathbf{x}\mathbf{x}^*] \tilde{\Phi} \Phi^* = \sigma^2 \Phi \tilde{\Phi}^* \tilde{\Phi} \Phi^*.$$

#### 4.7. Papoulis–Gerchberg Algorithm

Consider the inpainting problem discussed in Section 4.7.1.

- (i) Pose the problem as solving a linear system.
- (ii) Calculate condition numbers for the two cases of pixels missing in a block as opposed to missing randomly. Consider a one-dimensional vector  $\mathbf{x}$  of size  $N = 1024$ , with a bandlimited DFT that is zero for  $|K| > 400$ . In  $\mathbf{x}$ , set 128 samples to 0, in three different ways by zeroing out (a) the first 128 samples, (b) every 8th sample, and (c) random 128 samples. Compare the resulting condition numbers, and speed of convergence of the corresponding Papoulis–Gerchberg algorithms.

*Solution:* TBD.

## Exercises

### 4.1. Least-Squares Approximation of Sampling Followed by Interpolation

Let  $x(t) \in \mathcal{L}^2(\mathbb{R})$ . Show that

$$\min_{\hat{x}(t) \in S} \int_{-\infty}^{\infty} |x(t) - \hat{x}(t)|^2 dt,$$

where  $S$  is the space of functions piecewise constant over integer intervals, is obtained for

$$\hat{x}(t) = \sum_{n \in \mathbb{Z}} \left( \int_n^{n+1} x(t) dt \right) \chi_{[0,1)}(t-n).$$

Use this to show that  $P = \Phi\Phi^*$  with  $\Phi^*$  the sampling operator from (4.2), and  $\Phi$  the interpolation operator from (4.3), achieves this least-squares approximation.

### 4.2. Sampling and Interpolation for Bandlimited Vectors

Given is a vector  $x \in \mathbb{C}^M$  and its DFT  $X$  from (2.159a). It is said to have *bandwidth*  $k_0$ ,  $k_0$  odd,<sup>88</sup> when its DFT satisfies

$$X_k = 0 \quad \text{for all } \left| k - \frac{M}{2} \right| < \frac{k_0 - 1}{2}.$$

Define the subspace of bandlimited vectors  $\text{BL}\{-k_0 \bmod M, \dots, k_0 \bmod M\}$  when  $x$  that belongs to that subspace has bandwidth  $k_0$ . For  $x$  in such a bandlimited subspace, find  $\Phi$  so that the system described by  $\Phi\Phi^*$  in Section 4.2.1 achieves perfect recovery  $\hat{x} = x$ .

### 4.3. Orthogonal Projection with No Sampling Prefilter

Consider the system depicted in Figure 4.14(b) with no sampling prefilter. Under what condition on the interpolation postfilter  $g$  does the system implement an orthogonal projection? Describe the subspace that is the range of that orthogonal projection.

### 4.4. Sampling the DTFT of a Finite-Length Sequence

Consider a real sequence  $x_n$  with finite support  $[-(k_0 - 1)/2, (k_0 - 1)/2]$ . Show that its DTFT  $X(e^{j\omega})$  can be sampled at  $k_0$  points,  $\omega_k = 2\pi k/k_0$ , or  $X_k = X(e^{j\omega_k})$  so that  $x_n$  can be recovered from  $X_k$ .

### 4.5. Bandlimiting as Orthogonal Projection

- (i) Show that an ideal lowpass filter with cut-off frequency  $\omega_0/2$  computes the orthogonal projection of its input onto  $\text{BL}[-\omega_0/2, \omega_0/2]$ .
- (ii) Indicate the class of filters that compute orthogonal projections (not necessarily lowpass).

### 4.6. Bandlimited space with rational sampling rate changes

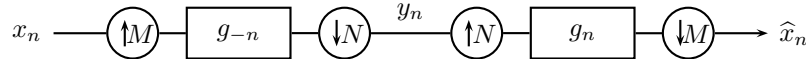
Given is a system in Figure P4.6-1 and a sequence  $x_n \in \text{BL}[-2\pi/3, 2\pi/3]$ . Find out what the filter  $g_n$  has to be for  $\hat{x} = x$  when:

- (i)  $M = 2, N = 3$ ;
- (ii) general coprime  $M$  and  $N$ .

### 4.7. Bases for Shift-Invariant Subspaces

Let  $x_n = \delta_n + \delta_{n-1} + \delta_{n-2}$  and let  $S$  be the shift-invariant subspace with respect to shift 2 generated by  $x$ . Find an orthonormal basis for  $S$ . What up- and downsampling factor, sampling prefilter and interpolation postfilter should be used so that the system in Figure 4.17(b) computes an orthogonal projection to  $S$ ?

<sup>88</sup> $k_0$  is odd because for a real spectrum,  $X_k = X_{-k}^*$ , see Table 3.4.



**Figure P4.6-1:** Sampling followed by interpolation with rational sampling rate changes.

4.8. *Ideally Matched Sampling and Interpolation with Nonorthogonal Sequences*

Find the ideally matched

- (i) interpolation operator for the sampling operator  $\tilde{\Phi}^*$  in (4.37a); as well as
- (ii) sampling operator for the interpolation operator  $\Phi$  in Example 4.12.

4.9. *Western Movies and Sampling*

Classic movies were shot with 24 frames/s, each frame being a snapshot of the scene, with an exposure time of roughly 10 ms. Given a carriage with wagon wheels having 16 spokes of diameter 2 m, what is/are the speed(s) so that the wheels look motionless?

4.10. *Downsampling by N*

Prove the expression for downsampling by  $N$ , (2.184), by going back to the underlying continuous-time function  $x_c(t)$  and resampling it with an  $N$ -times longer sampling period. That is, consider  $x_n$  and  $y_n = x_{nN}$  as two sampled versions of the same continuous-time function  $x_c(t)$ , with sampling periods  $T$  and  $NT$ , respectively. (Hint: Use (4.51) and (4.50).)

4.11. *Multirate System*

For the discrete system  $y_n = (U_3GD_2x)_n$ , with  $G$  an LSI filter:

- (i) Give the  $z$ -transform of the output signal.
- (ii) Assume the underlying continuous-time input function  $x_c(t) \in \text{BL}[-\pi/T, \pi/T]$ , and that it was sampled at  $1/T$  Hz. Write  $Y(e^{j\omega})$  as a function of  $X_c(\omega)$  and  $G(e^{j\omega})$  in the frequency domain. Also, specify the increase/decrease in the sampling rate achieved by this system and the conditions on  $X_c(\omega)$  to avoid aliasing.

4.12. *Sinc Orthonormal Basis for  $\text{BL}[-\pi/T, \pi/T]$*

Let  $T \in \mathbb{R}^+$ , and for each integer  $n$  let

$$\varphi_n(t) = \frac{1}{\sqrt{T}} \text{sinc}\left(\frac{\pi}{T}(t - nT)\right).$$

- (i) Show that the family  $\{\varphi_n(t)\}_{n \in \mathbb{Z}}$  is orthonormal, that is,

$$\langle \varphi_n(t), \varphi_m(t) \rangle = \delta_{n-m}.$$

- (ii) Show that this family forms an orthonormal basis for  $\text{BL}[-\pi/T, \pi/T]$ .

4.13. *Bandpass Sampling*

Refer to Example 4.16 and the spectrum  $X(\omega)$  in (4.59).

- (i) Show that the solution described in the example can be expressed as an orthonormal expansion for the bandpass subspace  $\text{BL}([-3\pi, -2\pi] \cup [2\pi, 3\pi])$  and give the basis functions.
- (ii) An alternative to a pure sampling solution is to demodulate (shift in frequency) the bandpass signal into the baseband, or  $\text{BL}[-\pi, \pi]$ . Give the demodulation function, the sampling, and the reconstruction (which requires modulation).
- (iii) Show that bandpass signals with frequency support  $[-(K+1)\omega_0/2, -K\omega_0/2] \cup [K\omega_0/2, (K+1)\omega_0/2]$  can be sampled with sampling frequency  $\omega_s = \omega_0$ , and perfectly reconstructed.

4.14. *Continuous-Time Modulation Using Discrete-Time Operators*

In Proposition 4.16, we saw how continuous-time convolution can be implemented in discrete time. Consider two bandlimited functions,  $x(t) \in \text{BL}[-\omega_x/2, \omega_x/2]$  and  $g(t) \in \text{BL}[-\omega_g/2, \omega_g/2]$ .

- (i) Show how the multiplication  $y(t) = g(t)x(t)$  can be implemented in discrete time.
- (ii) Compute  $y(t)$  if  $x(t)$  is as in (4.56) and  $g(t)$  is a cosine function as in (4.55a) with  $\omega_0 = 3\pi$ .
- (iii) Is there a more efficient way to compute  $y(t)$  in ((ii))?  
(Hint: Use the digital-to-analog conversion and interpolation.)

4.15. *Multichannel Sampling*

Let  $x(t) \in \text{BL}[-\pi, \pi]$ .

- (i) Let  $N = 3$  and show how 3 channels followed by sampling with period  $T = 3$  perfectly reconstruct the input function if and only if the matrix in (4.67) is nonsingular for  $\omega \in [0, 2\pi/3]$ .
- (ii) Repeat (i), but where the channel inputs are  $x(t)$ ,  $x'(t)$ , and  $x''(t)$ .
- (iii) Generalize (ii) to an arbitrary number of derivatives, that is, prove that an  $N$ -channel filter banks with channel inputs  $x(t)$ ,  $x'(t)$ ,  $\dots$ ,  $x^{(N-1)}(t)$ , each sampled with period  $T = N$ , perfectly reconstructs the input function if and only if the matrix in (4.67) is nonsingular for  $\omega \in [0, 2\pi/N]$ .
- (iv) Consider irregular sampling and again a 3-channel system. Let  $\tilde{g}_0(t) = \delta(t)$ ,  $\tilde{g}_1(t) = \delta(t - 1 - \alpha)$  and  $\tilde{g}_2(t) = \delta(t - 1 - \beta)$ , where  $\alpha, \beta \in [-1, 1]$ . For which values of  $\alpha, \beta$  does  $\det(\tilde{G}(\omega)) = 0$ ? Plot the condition number of  $\tilde{G}(\omega)$  for  $\alpha = 1$ ,  $\beta \in [-1/2, 1/2]$  and  $\omega \in [0, 2\pi/3]$ . Comment on the result.

4.16. *Adjoint Operators*

Prove that the sampling operator (4.78) and the interpolation operator (4.80) are adjoints of each other.

(Hint: Mimic the proof for functions, (4.45).)

4.17. *Dirac Delta Comb Fourier Series Pair*

Prove that (4.82a) is indeed a Fourier-series transform pair.

4.18. *Dirichlet Kernel*

In (4.90) we computed the Dirichlet kernel by assuming a bandlimited periodic function and a periodic filter  $g(t)$ . Do the same by assuming a nonperiodic filter  $g(t)$  and then sampling the result in Fourier domain.

4.19. *Fourier Series with Triangle Spectrum*

Let  $x(t) \in \text{BL}\{-k_h, \dots, k_h\}$ , with  $k_s = 2k_h + 1$ , be a periodic function with period  $T = 1$  whose Fourier series coefficients are real and symmetric and given by

$$X_k = \begin{cases} 1 - \frac{|k|}{k_h}, & |k| \leq k_h; \\ 0, & \text{otherwise.} \end{cases}$$

- (i) Show that sampling  $x(t)$  at  $nT_s = n/k_s$ ,  $k = -k_h, k_h + 1, \dots, k_h - 1, k_h$ , ensures perfect function recovery and indicate how to do it.
- (ii) Comment on what happens when sampling uniformly with fewer than  $k_s$  samples.

4.20. *Continuous-Time Function Leading to i.i.d. Samples*

Let  $x(t)$  be continuous-time function whose projection onto  $\text{BL}[-\pi/T, \pi/T]$  is a continuous-time white Gaussian noise  $\hat{x}(t)$ . Prove that filtering  $x(t)$  with an ideal lowpass filter of bandwidth  $\omega_0 = 2\pi/T$  and sampling with period  $T$  leads to a sequence of samples  $y_n$  that is discrete-time white Gaussian noise.

## Chapter 5

# Approximation and Compression

## Contents

5.1	Introduction . . . . .	456
5.2	Approximation of Functions on Finite Intervals by Polynomials . . . . .	461
5.3	Approximation of Functions by Splines . . . . .	480
5.4	Approximation of Functions and Sequences by Series Truncation . . . . .	493
5.5	Compression and Transform Coding . . . . .	500
5.6	Computational Aspects . . . . .	513
	Chapter at a Glance . . . . .	514
	Historical Remarks . . . . .	515
	Further Reading . . . . .	516
	Exercises with Solutions . . . . .	516
	Exercises . . . . .	520

In previous chapters, we saw how to write a sequence or a function as an expansion in a basis (Chapters 1–3), or using sampling and interpolation (Chapter 4). Often, however, we do not have the luxury of representing the sequence or function exactly, requiring the development of approximate representations:

- (i) If the expansion is too big, that is, it requires too many coefficients to represent the function or the sequence, we need truncation methods. For example, in a Fourier series representation, we can decide to keep a subset of the coefficients. How many and which ones to keep influence the choice of the approximation method, and the errors so incurred affect the approximation quality.
- (ii) If the function varies too fast, then sampling cannot catch the variations fast enough. We then typically first smooth (bandlimit) the function, which amounts to projecting onto a subspace, such as lowpass filtering we saw in Section 4.4.2. The projection error is a measure of how well we can approximate the function using a finite bandwidth approximation.

- (iii) If the function or sequence is too complex, that is, the expansion coefficients require too much storage space or bandwidth for transmission, we need to both reduce their number as in (i), as well as their accuracy, leading to compression.

Approximation theory deals with the choice of expansion coefficients to keep; compression theory deals with approximating those coefficients. Some methods are very classical, such as approximation via Taylor series, while others are more recent, such as nonlinear approximation in bases. We review a collection of approaches with a clear message: there is no *one-size-fits-all* technique. Each case requires a careful analysis and a well-engineered solution. When this is properly done, a good solution can have high impact, as demonstrated by the billions of cameras, mobile phones and computers using JPEG to represent images efficiently.

## 5.1 Introduction

Given a function  $x(t)$ , we often do not have the luxury of representing it exactly, but must instead be satisfied with an approximation  $\hat{x}(t)$ . This may be due to a variety of reasons.

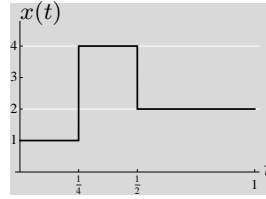
- (i) The function may not be known everywhere, but only at a number of specific points. Given these points, we may try to interpolate the function as well as we can.
- (ii) The function may be known everywhere, and may thus have an expansion such as a Fourier series over an interval, but we may not be able to afford to compute the full expansion for complexity and storage reasons. Instead, we may be able to obtain just an approximate expansion or a projection on a subspace. Then, we may be interested in knowing how good the approximation is.
- (iii) Finally, expansion coefficients themselves, which are typically real numbers, might have to be approximated as well, given the finite precision and storage requirements. This is when we consider signal compression.

Of these various approximations, we have already discussed the projection onto a subspace in the particular case of sampling and interpolation in the previous chapter. Our aim here is to broaden this approximation view to a wider range of methods, including compression. To make these points concrete, we go through a simple example.

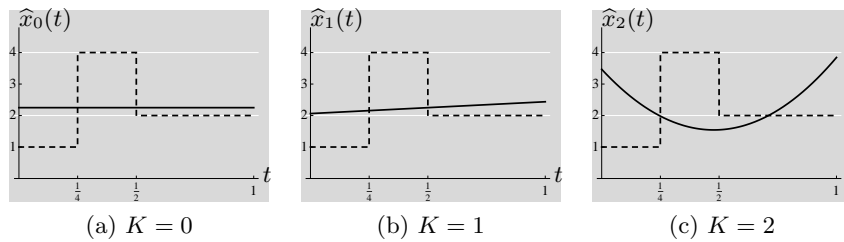
**Function to Be Approximated** Given are piecewise constant functions over the unit interval. Assume the number of constant pieces  $K$  to be known, while the transition points  $\{t_k\}_{k=1}^{K-1}$  and the values of the constants  $\{\alpha_i\}_{i=0}^{K-1}$  are unknown but in the interval  $[0, 1]$ . An example function for  $K = 3$  is shown in Figure 5.1. Such a function is called parametric since, given  $K$ , the set of  $2K - 1$  parameters  $\{t_k\}_{k=1}^{K-1}$  and  $\{\alpha_i\}_{i=0}^{K-1}$  specify  $x(t)$ .

Let us consider a few possible approximate representations.





**Figure 5.1:** Piecewise constant function with  $K = 3$ ,  $\{t_1, t_2\} = \{1/4, 1/2\}$ ,  $\{\alpha_0, \alpha_1, \alpha_2\} = \{1, 2, 4\}$ .

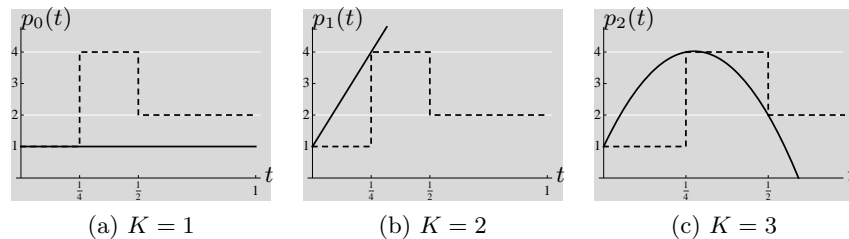


**Figure 5.2:** Best least-squares approximation  $\hat{x}_K(t)$  of  $x(t)$  from Figure 5.1 using a Legendre polynomial basis of degree  $K$ .

**Least-Squares Polynomial Approximation** First, we try fitting  $x(t)$  with a polynomial of degree  $L$ . We should not expect too much, since the function is discontinuous, while polynomials are smooth. As an example, take Legendre polynomials seen in Solved Exercise 1.5, since they form an orthonormal basis for an interval. In Figure 5.2, we show the best least-squares approximation  $\hat{x}_K(t)$  using Legendre polynomials of degree  $K = 0, 1, 2$ .

**Lagrange Interpolation: Matching Points** Instead of using orthogonal polynomials such as Legendre polynomials, we can try polynomial interpolation where specific points of the function are exactly matched, as is the case, for example, with Lagrange interpolation. An advantage is that the function need not be known everywhere, but only at the interpolation points, which must be distinct. Figure 5.3 show the result for 1, 2, 3 known points from Figure 5.1. The result is not entirely satisfactory, unsurprisingly so, since polynomials are necessarily smooth, unlike the function we are trying to match.

**Taylor Series Expansion: Matching Derivatives** When an analytical expression of the function to be approximated is available and derivatives exist up to some order, we can match the function and its derivatives using the well-known Taylor series at a point of interest. The advantage is that the representation is exact at that point, while it gradually worsens when moving away. In some sense, one can think of Taylor series as function extrapolation, as opposed to Lagrange method, which is a function-interpolation method.



**Figure 5.3:** Lagrange interpolation of  $x(t)$  from Figure 5.1 using  $K$  known distinct points.

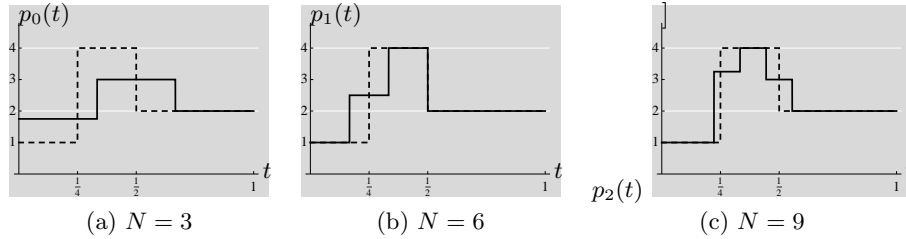
**Other Polynomial Approximation Methods** A mixture of Lagrange interpolation and Taylor extrapolation is a hybrid method called Hermite interpolation, which matches both points and derivatives. While Legendre polynomials minimize the quadratic approximation error, Lagrange, Taylor and Hermite methods minimize a particular norm. Minimizing the maximum error leads to minimax polynomial approximation and Chebyshev polynomials.

**Approximation of Functions by Splines** Polynomial approximation is inherently local. For a function on the real line, while approximating by polynomials on a set of successive intervals is a possibility, we will encounter problems at interval boundaries. Instead, methods preserving continuity and derivatives are preferable. Among such methods, splines are the most popular, partially due to their computational efficiency. We consider in particular uniform splines, since they generate a shift-invariant subspace and are closely related to regular sampling.

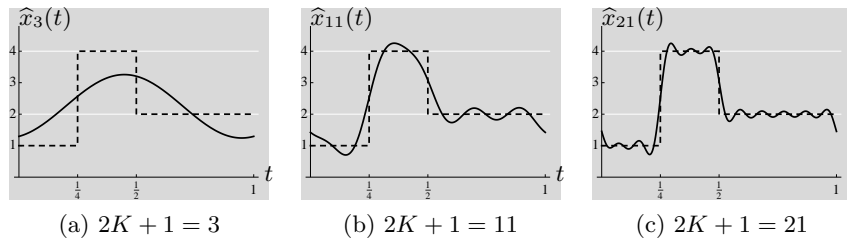
Consider approximating our function in Figure 5.1 using constant splines, or elementary box function (3.76). This seems like a good idea because both the constant spline and the function we want to approximate are piecewise constant. However, the constant spline will have a support of length  $1/N$  when the interval  $[0, 1]$  is split into  $N$  uniform pieces. In Figure 5.4, we show the approximation using 3, 6 and 9 constant splines. The good news is that the resulting approximation is piecewise constant, the not so good news is that to be close to the original function, the approximation requires  $N$  to be large. Because of the particular choice of transition points in the function from Figure 5.1, choosing  $N = 4$  would have given a perfect approximation. Since constant splines and their shifts form an orthogonal basis, this approximation is a best least-squares approximation as well.

Note that splines, in particular uniform splines of which the constant function is the lowest-order representative, have a number of nice properties that will be explored in more detail in this chapter. Functions living in spline spaces are another example (besides bandlimited functions and sampling) where discrete-time processing performs continuous-time processing in a precise way, and we will show this specifically for derivatives and integrals.

**Linear Approximation in Fourier Bases** Since we are considering a function on an interval, it is natural to look at the Fourier series representation. We already



**Figure 5.4:** Length-1/N constant spline approximation of  $x(t)$  from Figure 5.1.

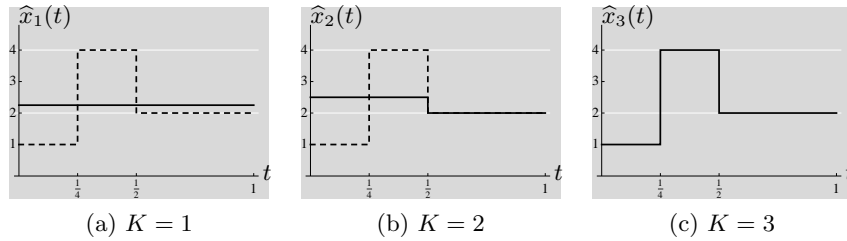


**Figure 5.5:** Linear approximation of  $x(t)$  from Figure 5.1 using the Fourier series with  $2K + 1$  lowest-frequency terms.

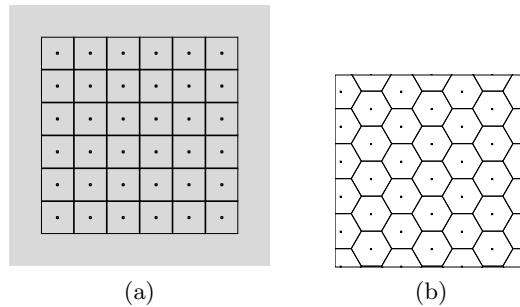
know that the Gibbs phenomenon will hurt at discontinuities, and lead to poor convergence. Figure 5.5 shows the Fourier series approximation with  $2K + 1$ , with  $K = 1$ ,  $K = 5$ , and  $K = 10$  lowest-frequency terms. Since the Fourier series is an orthonormal system, we obtain again a best least-squares approximation, but the trigonometric polynomials of the Fourier series are not a good match to the piecewise constant function and convergence is slow.

The approximation we just saw is an example of linear approximation accomplished via orthogonal projection. We decide a priori that  $\hat{x}(t)$  is the orthogonal projection of  $x(t)$  onto a (fixed) subspace of the lowest  $2K + 1$  Fourier series basis vectors.

**Nonlinear Approximation** An alternative is to choose the largest-magnitude coefficients in the orthonormal basis, which now depends on the particular  $x(t)$  we wish to approximate. This is an *adaptive subspace approximation*, because we choose the *best* subspace depending on the function to be approximated. In Figure 5.6, we show this approximation using an orthonormal Haar wavelet basis, and retaining the largest coefficients in the expansion. As can be seen, the approximation is much better than that with Fourier series. This is due to both the basis (Haar wavelets, being piecewise constant, are better suited to the function we wish to approximate) as well as the approximation method (adapting to the function by retaining the largest coefficients). We explore both linear and nonlinear approximation in orthonormal bases in what follows. In the case of random processes, a classic linear approximation method is the *Karhunen-Loève transform (KLT)*, a principal component analysis based on the autocorrelation matrix of the process.



**Figure 5.6:** Nonlinear approximation of  $x(t)$  from Figure 5.1 in the Haar wavelet basis with  $K$  largest terms.

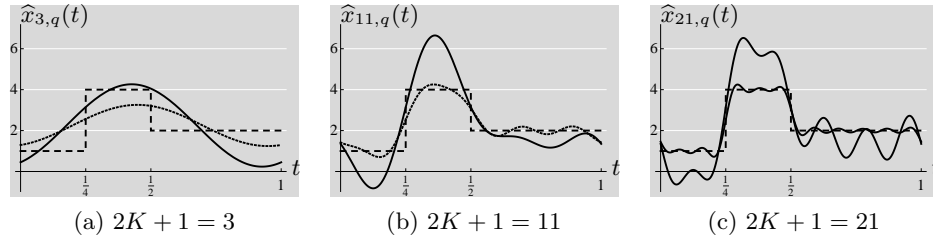


**Figure 5.7:** Quantization of the unit square with (a) square and (b) hexagonal cells.

**Compression** Finally, let us turn to compression, where given a representation (could be an approximation using a polynomial or a basis), we need to approximate it, or quantize the parameters of the representation. The simplest form of this approximation is *rounding to the nearest integer*. Consider the unit square in  $\mathbb{R}^2$ ,  $[0, 1]^2$ . The rounding-to-the-nearest-integer rule means that all vectors inside a given cell are represented by that cell's center, shown in Figure 5.7(a) with square cells and Figure 5.7(b) with hexagonal cells.

When a representation is to be stored on a computer or transmitted over a channel, the number of bits used to describe the signal becomes the currency. Take the Fourier series representation, with 3 bits for each of the coefficients, then 3, 33, and 63 bits are used for the various linear approximations in Figure 5.5. This results in an increased error; Figure 5.8 shows the result. In addition, the various binary numbers may have different probabilities of occurring, or various entropies, which potentially leads to more compression.

What we just saw raises many questions which are at the heart of signal compression, such as (1) what the best representation for compression purposes is; (2) what good quantization strategies are; and (3) how one should allocate bits in a representation. The answers will depend on signal models and the related notion of entropy. A branch of information theory called rate-distortion theory gives some fundamental bounds in a somewhat abstract setting where complexity is not an issue. In the more practical realm of real signal compression, transform coding has become the standard answer as well as the workhorse of multimedia compression



**Figure 5.8:** Linear approximation (Figure 5.5, dotted) of  $x(t)$  (Figure 5.1, dashed) using the quantized Fourier series (solid) with  $2K + 1$  lowest-frequency terms and 3 bits/coefficient.

algorithms. This has led to methods with high practical impact, such as audio compression as in MP3 or image compression as in JPEG.

### Chapter Outline

The chapter follows this brief introduction: Section 5.2 discusses approximation of functions on finite intervals by polynomials, starting with the orthonormal basis given by Legendre polynomials, moving to matching a function at specific points using Lagrange interpolation, and reviewing the classic Taylor expansion that matches a function and its derivatives at a given point. We discuss minimax approximation, in particular using Chebyshev polynomials. Section 5.3 looks into approximating functions on the real line by splines and the shift-invariant subspace they generate. We calculate explicit projections on spline spaces, as well as orthogonalizations. We also calculate continuous-time operators such as derivatives and integrals on the discrete sequence of spline coefficients. Section 5.4 considers approximations in bases, both with linear and nonlinear methods via truncation of coefficients. The linear approximation method just truncates the series, while the nonlinear one is an adaptive subspace selection matched to the function to be approximated; the coefficients are first reordered and then truncated. For stochastic processes, the Karhunen–Loève transform (KLT) is derived as an optimal linear approximation method. In Section 5.5, we switch gears and approximate expansion coefficients, leading to compression and transform coding. We review entropy coding, quantization and bit allocation. Then, we discuss transform coding in detail, together with its main components. Section 5.6 closes with computational aspects.

## 5.2 Approximation of Functions on Finite Intervals by Polynomials

The previous chapter focused on sampling and interpolation operators and both exact as well as approximate representations of sequences defined on integers and functions defined on the real line using these operators. The infinite lengths encountered there made it imperative to use LSI filtering in the sampling and interpolation operations. Otherwise, similarly to what we saw with nonuniform sampling,

the computations can become complicated. We now shift our attention to approximating a function on a finite interval using polynomials. Here, we will not have shift-invariance properties, and, in fact, the behavior near the endpoints of the interval is often quite different than the behavior near the center.

Throughout this section, we denote by  $x(t)$  the function to be approximated on the finite interval  $[a, b] \in \mathbb{R}$ , the approximating polynomial of degree at most  $K$  (see (2.280)) by

$$p_K(t) = \sum_{k=0}^K \alpha_k t^k, \quad (5.1a)$$

and the error between the function  $x(t)$  and its approximation  $p_K(t)$  by

$$\epsilon_K(t) = x(t) - p_K(t). \quad (5.1b)$$

Minimizing the appropriate norm of the error leads to different types of approximations; bounds on such norms gives insight into the quality of the approximation.

We start in Section 5.2.1 with least-squares polynomial approximation, in which series expansions with respect to orthogonal polynomials in general, and Legendre polynomials in particular, arise naturally. We then discuss matching approximating polynomials to a given function at a specific number of points, Lagrange interpolation in Section 5.2.2, or matching derivatives up to a certain order at a single point, Taylor series expansion in Section 5.2.3; Hermite interpolation in Section 5.2.4 combines the two. Minimizing  $\mathcal{L}^\infty$  error is first studied in Section 5.2.5 and then applied to FIR filter design.

### 5.2.1 Least-Squares Approximation

Let  $x(t)$  be a real-valued function in  $\mathcal{L}^2([a, b])$ . An approximation  $\hat{x}$  that minimizes

$$\|x - \hat{x}\|_2^2 = \int_a^b (x(t) - \hat{x}(t))^2 dt$$

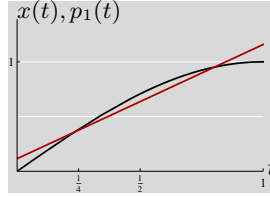
over some set of possibilities for  $\hat{x}$ , is called a *least-squares* approximation. Least-squares approximation of such a function in the subspace  $\mathcal{P}_K[a, b]$  spanned by  $p_K(t)$ , polynomials of degree at most  $K$ , is straightforward using the tools developed in Chapter 1. Consider the  $\mathcal{L}^2$  norm of the error between  $x$  and  $p_K$ ,<sup>89</sup>

$$\|\epsilon_K\|_2^2 = \|x - p_K\|_2^2 = \int_a^b (x(t) - p_K(t))^2 dt.$$

Because of the projection theorem, Theorem 1.26, the solution to this approximation problem is the orthogonal projection of  $x$  onto  $\mathcal{P}_K[a, b]$  given an orthonormal basis for  $\mathcal{P}_K[a, b]$ . Calling  $\{\varphi_0(t), \varphi_1(t), \dots, \varphi_K(t)\}$  such a basis, the least-squares approximation is (see Theorem 1.40)

$$p_K(t) = \sum_{k=0}^K \langle x, \varphi_k \rangle \varphi_k(t).$$

<sup>89</sup>We explicitly denote here the  $\mathcal{L}^2$  norm, but will drop it when there is no risk of confusion.



**Figure 5.9:** Least-squares approximation of  $x(t) = \sin \pi t/2$  on  $[0, 1]$ .

This is a polynomial of degree at most  $K$  because each of the  $\varphi_k$ s is a polynomial of degree at most  $K$ .

**EXAMPLE 5.1 (LEAST-SQUARES APPROXIMATION)** Let us find the least-squares degree-1 polynomial approximation of  $x(t) = \sin \pi t/2$  on  $[0, 1]$ . For this, we need an orthonormal basis for  $\mathcal{P}_1[0, 1]$ ; one such basis is  $\{1, \sqrt{3}(2t - 1)\}$ .<sup>90</sup> The least-squares approximation is

$$\begin{aligned} p_1(t) &= \langle \sin \frac{\pi t}{2}, 1 \rangle \cdot 1 + \langle \sin \frac{\pi t}{2}, \sqrt{3}(2t - 1) \rangle \sqrt{3}(2t - 1) \\ &= \frac{2}{\pi} + \frac{2\sqrt{3}(4 - \pi)}{\pi^2} \sqrt{3}(2t - 1) = \frac{12(4 - \pi)}{\pi^2} t + \frac{8(\pi - 3)}{\pi^2}. \end{aligned}$$

This approximation is illustrated in Figure 5.9.

The method above uses an orthonormal basis for  $\mathcal{P}_K[a, b]$ . Such an orthonormal basis is not unique. However, if one seeks a single sequence of vectors  $\{\varphi_0, \varphi_1, \dots\}$  such that, for every  $K \in \mathbb{N}$ ,

$$\{\varphi_0, \varphi_1, \dots, \varphi_K\} \text{ is an orthonormal basis for } \mathcal{P}_K[a, b],$$

the solution is unique up to multiplications by  $\pm 1$  and is obtained by using the Gram–Schmidt procedure on  $\{1, t, t^2, \dots\}$ . This construction creates an orthonormal basis for  $\mathcal{P}_K[a, b]$  that is the union of an orthonormal basis for  $\mathcal{P}_{K-1}[a, b]$  and  $\varphi_K$ . With this nested sequence of bases, the least-squares approximations satisfy a recursion:

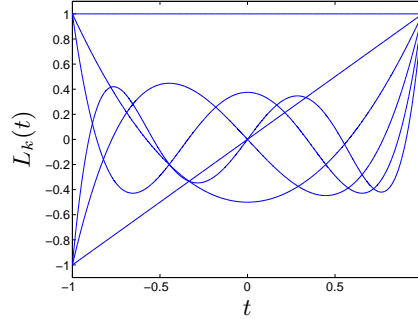
$$p_K = p_{K-1} + \langle x, \varphi_K \rangle \varphi_K. \quad (5.2)$$

**Legendre Polynomials** A known class of polynomials that can be used for least-squares approximation is the class of *Legendre polynomials*,

$$L_k(t) = \frac{(-1)^k}{2^k k!} \frac{d^k}{dt^k} (1 - t^2)^k, \quad k \in \mathbb{N}, \quad (5.3)$$

orthogonal on  $[-1, 1]$ . The first few are shown in Figure 5.10 and tabulated below:

$$\begin{aligned} L_0(t) &= 1 & L_3(t) &= \frac{1}{2}(5t^3 - 3t) \\ L_1(t) &= t & L_4(t) &= \frac{1}{8}(35t^4 - 30t^2 + 3) \\ L_2(t) &= \frac{1}{2}(3t^2 - 1) & L_5(t) &= \frac{1}{8}(63t^5 - 70t^3 + 15t) \end{aligned}$$



**Figure 5.10:** The first six Legendre polynomials (up to degree 5),  $\{L_k\}_{k=0}^5$ .

Legendre polynomials are orthogonal but not orthonormal; an orthonormal set can be obtained by dividing by the norms  $\|L_k\| = \sqrt{2/(2k+1)}$ . The first few in normalized form are:

$$\begin{aligned} \bar{L}_0(t) &= \frac{1}{\sqrt{2}} & \bar{L}_3(t) &= \frac{\sqrt{7}}{2\sqrt{2}}(5t^3 - 3t) \\ \bar{L}_1(t) &= \frac{\sqrt{3}}{\sqrt{2}}t & \bar{L}_4(t) &= \frac{3}{8\sqrt{2}}(35t^4 - 30t^2 + 3) \\ \bar{L}_2(t) &= \frac{\sqrt{5}}{2\sqrt{2}}(3t^2 - 1) & \bar{L}_5(t) &= \frac{\sqrt{11}}{8\sqrt{2}}(63t^5 - 70t^3 + 15t) \end{aligned}$$

Once we have the Legendre polynomials, changing the interval of interest does not require a tedious application of the Gram–Schmidt procedure as we have done in Solved Exercise 1.5. Instead, polynomials orthogonal with respect to the  $\mathcal{L}^2$  inner product on  $[a, b]$  can be found by shifting and scaling the Legendre polynomials; see Exercise 5.1.

**EXAMPLE 5.2 (LEAST-SQUARES APPROXIMATION WITH LEGENDRE POLYNOMIALS)**

Let  $x(t) = t \sin 5t$ . To form least-squares polynomial approximations on  $[0, 1]$ , we need orthogonal polynomials in  $\mathcal{L}^2([0, 1])$ ; we can obtain these by shifting and scaling the Legendre polynomials. The first few, in normalized form, are:

$$\begin{aligned} \varphi_0(t) &= 1 & \varphi_2(t) &= \sqrt{5}(6t^2 - 6t + 1) \\ \varphi_1(t) &= \sqrt{3}(2t - 1) & \varphi_3(t) &= \sqrt{7}(20t^3 - 30t^2 + 12t - 1) \end{aligned}$$

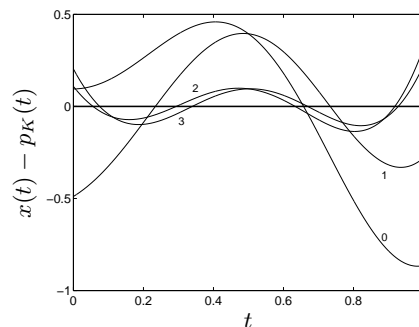
The best constant approximation is  $p_0 = \langle x, \varphi_0 \rangle \varphi_0$ , and higher-degree least-squares approximations can be found through (5.2). The least-squares approximations up to degree 3 are:

$$\begin{aligned} p_0(t) &\approx -0.0951 & p_2(t) &\approx -0.109 + 2.42t - 3.59t^2 \\ p_1(t) &\approx 0.489 - 1.17t & p_3(t) &\approx -0.204 + 3.56t - 6.43t^2 + 1.90t^3 \end{aligned}$$

The resulting approximation errors are shown in Figure 5.11.

<sup>90</sup>This particular basis comes from applying the Gram–Schmidt procedure to  $(1, t)$ .





**Figure 5.11:** Errors of least-squares polynomial approximations of  $x(t) = t \sin 5t$  on  $[0, 1]$ , for degrees  $K = 0, 1, 2, 3$ . Curves are labeled by the polynomial degree.

**Orthogonal Polynomials** While Legendre polynomials and their shifted versions solve our least-squares approximation problems, changing the inner product to

$$\langle x, y \rangle = \int_a^b x(t)y(t)W(t)dt,$$

where  $W(t)$  is a nonnegative *weight function*, would change orthogonality relationships among polynomials. One of the ramifications is that applying the Gram–Schmidt procedure to the ordered monomials  $\{1, t, t^2, \dots\}$  would yield a different set of polynomials. Also, if the weight function has adequate decay, inner products between polynomials on  $[a, \infty)$  or  $(-\infty, \infty)$  can be finite, so we need not consider only finite intervals. In *Chapter at a Glance*, we give definitions, weight functions and intervals of orthogonality for different families of polynomials.

Orthogonal polynomials have many applications in approximation theory and numerical analysis; Exercises 5.2 and 5.3 explore some of their properties. We will consider Chebyshev polynomials in Section 5.2.5 because they play an important role in  $\mathcal{L}^\infty$  approximation.

### 5.2.2 Lagrange Interpolation: Matching Points

In many situations in which we desire a polynomial approximation of a function on an interval, we cannot use inner products of the function with a set of basis functions because we may not know the function on the entire interval, or we may not want to compute the required integrals.<sup>91</sup> However, if we know the values of the function at certain points in the interval, we can possibly use polynomial matching of the function at these points. We now look at when such matching exists, when it is unique, and how to bound the approximation error.

From (5.1a), a polynomial of degree  $K$  has  $K + 1$  parameters and is thus determined by  $K + 1$  independent and noncontradictory constraints. Let us look carefully at whether specifying  $p_K$  at  $K + 1$  points provides suitable constraints.

<sup>91</sup>The difficulty of computing integrals may be the reason for forming a polynomial approximation, leading to numerical integration techniques such as the trapezoidal rule and Simpson’s rule.

Fix a set of  $K+1$  distinct points  $\{t_k\}_{k=0}^K \subset [a, b]$  called *nodes*, and assume that the values of the function  $x$  at the nodes,  $y_k = x(t_k)$ ,  $k = 0, 1, \dots, K$ , are known. Requiring  $p_K$  to match  $x$  at the nodes gives a system of  $K+1$  linear equations with  $K+1$  unknowns,

$$\begin{bmatrix} 1 & t_0 & t_0^2 & \dots & t_0^K \\ 1 & t_1 & t_1^2 & \dots & t_1^K \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & t_K & t_K^2 & \dots & t_K^K \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_K \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_K \end{bmatrix}. \quad (5.4)$$

The matrix in (5.4) is a square Vandermonde matrix, and is invertible if and only if the nodes are distinct; see (1.231). Thus, the values of  $x(t)$  at  $K+1$  distinct nodes uniquely specify an interpolation polynomial of degree at most  $K$ . As we know from Section 1.B, having more samples (equations) will not lead to a solution: the range of the Vandermonde matrix will be a proper subspace and the vector of samples will (in general) not be in that subspace. On the other hand, having fewer samples always leaves the polynomial unspecified, as there will be infinitely many solutions to (5.4).

To see this, suppose that  $p_{K-1}$  is the degree- $(K-1)$  polynomial uniquely specified by  $\{(t_k, y_k)\}_{k=0}^{K-1}$ . For any  $c \in \mathbb{R}$ , the degree- $K$  polynomial

$$p_K(t) = p_{K-1}(t) + c(t-t_0)(t-t_1)\cdots(t-t_{K-1})$$

will match  $p_{K-1}$  at all  $K-1$  nodes because the second term is zero at each node. Given the value at one additional node,  $y_K = x(t_K)$ , one can choose  $c$  so that  $p_K$  matches  $x$  all  $K$  nodes. This recursive process is one way to arrive at the general formula that follows.

**Lagrange Interpolation Formula** The unique polynomial of degree  $K$  that matches  $x$  at  $K+1$  distinct nodes  $\{t_k\}_{k=0}^K$  is given by

$$p_K(t) = \sum_{k=0}^K x(t_k) \prod_{\substack{i=0 \\ i \neq k}}^K \frac{t-t_i}{t_k-t_i}. \quad (5.5)$$

The polynomial  $p_K(t)$  is called the *Lagrange interpolating polynomial* for  $\{(t_k, x(t_k))\}_{k=0}^K$ . It interpolates correctly because

$$\prod_{\substack{i=0 \\ i \neq k}}^K \frac{t_\ell - t_i}{t_k - t_i} = \begin{cases} 1, & \text{for } \ell = k; \\ 0, & \text{otherwise,} \end{cases}$$

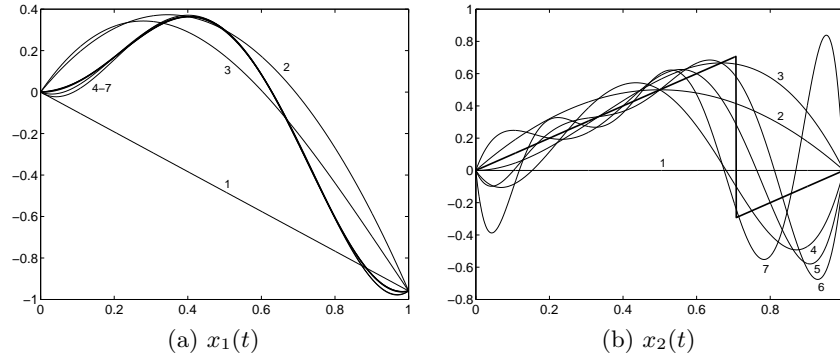
and the interpolating polynomial is unique.

**EXAMPLE 5.3 (LAGRANGE INTERPOLATION)** Let us construct approximations using (5.5) for two functions on  $[0, 1]$ , one continuous and the other not,

$$x_1(t) = t \sin 5t \quad \text{and} \quad x_2(t) = \begin{cases} t, & 0 \leq t < 1/\sqrt{2}; \\ t-1, & 1/\sqrt{2} \leq t < 1. \end{cases} \quad (5.6)$$

## 5.2. Approximation of Functions on Finite Intervals by Polynomials

467



**Figure 5.12:** Polynomial interpolation of functions in (5.6) for degrees  $K = 1, 2, \dots, 7$  and points taken uniformly on  $[0, 1]$ . Bold curves are the original functions and light curves are labeled by the polynomial degree.

Let the nodes be  $k/K$  for  $k = 0, 1, \dots, K$  (although evenly-spaced nodes are not a requirement). Figure 5.12 shows the functions (bold) and the interpolating polynomials for  $K = 1, 2, \dots, 7$ . The continuous function  $x_1$  is approximated much more closely than the discontinuous function  $x_2$ .

This example suggests that, when polynomial interpolation is used, the smoothness of a function affects the quality of approximation. Indeed, for functions that are sufficiently smooth over the range of interest (encompassing the nodes but also wherever the approximation is to be evaluated), the pointwise error can be bounded precisely using the following theorem:

**THEOREM 5.1 (ERROR OF LAGRANGE INTERPOLATION)** Let  $x$  have  $K + 1$  continuous derivatives on  $[a, b]$  for some  $K \geq 0$ , and let  $\{t_k\}_{k=0}^K \subset [a, b]$  be distinct. Then, with  $p_K$  defined in (5.5) and  $t \in [a, b]$ ,

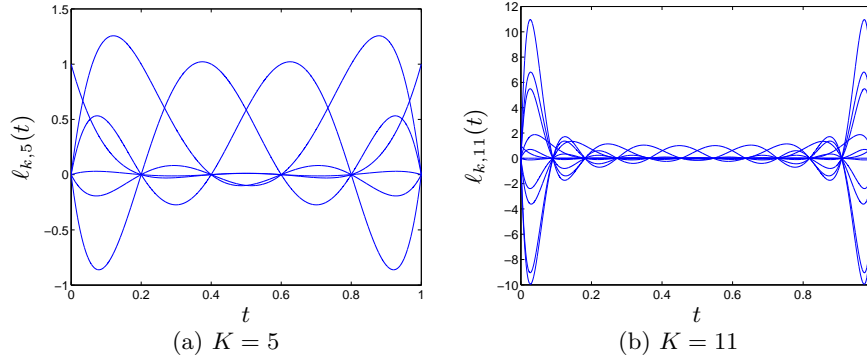
$$\epsilon_K(t) = \frac{\prod_{k=0}^K (t - t_k)}{(K + 1)!} x^{(K+1)}(\xi) \quad (5.7a)$$

for some  $\xi$  between the minimum and maximum of  $\{t, t_0, t_1, \dots, t_K\}$ . Thus,

$$|\epsilon_K(t)| \leq \frac{\prod_{k=0}^K |t - t_k|}{(K + 1)!} \max_{\xi \in [a, b]} |x^{(K+1)}(\xi)|, \quad (5.7b)$$

for every  $t \in [a, b]$ .

One immediate consequence of the theorem is that if  $x$  is a polynomial of degree  $K$ , the interpolant  $p_K$  matches everywhere. This follows from (5.7b) because  $x^{(K+1)}$  is identically zero. Of course, this was already obvious from the uniqueness of the interpolating polynomial. In general, the  $\max_{\xi \in [a, b]} |x^{(K+1)}(\xi)|$  factor in the



**Figure 5.13:** Bases of degree- $K$  Lagrange interpolating polynomials for  $K + 1$  nodes evenly spaced over  $[0, 1]$ :  $t_k = k/K$ ,  $k = 0, 1, \dots, K$ .

error bound is a global smoothness measure; it affects the pointwise error bound identically over the entire interval.

One interesting aspect of the error bound (5.7b) is that it depends on  $t$  through the  $\prod_{k=0}^K |t - t_k|$  factor. The error is zero at any node because of the interpolating property, but the bound behaves differently in the neighborhoods of different nodes. Moving away from a node by a small amount  $\varepsilon$ , that is, setting  $t = t_k + \varepsilon$ , we have

$$\prod_{k=0}^K |t - t_k| \approx |\varepsilon| \prod_{i, i \neq k} |t_k - t_i|.$$

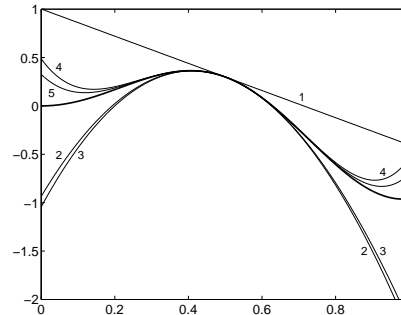
If the nodes are evenly spaced, the error bound becomes worse more quickly around an extremal node than around a central node (see also Figure 5.12). The potential for worse behavior at the endpoints can also be seen if we view the Lagrange interpolation formula as a series expansion using  $K + 1$  polynomials

$$\ell_{k,K}(t) = \prod_{\substack{i=0 \\ i \neq k}}^K \frac{t - t_i}{t_k - t_i}, \quad k = 0, 1, \dots, K, \quad (5.8)$$

as a basis. Each basis function depends on all of the nodes; two examples with evenly-spaced nodes are shown in Figure 5.13. These illustrate that at node  $t_k$ , basis function  $\ell_{k,K}$  is 1 and the other basis functions are 0. Also,  $\ell_{k,K}(t)$  is not necessarily small far away from  $t_k$ . This means that the sample  $x(t_k)$  affects the interpolation far from  $t_k$ . In particular, Figure 5.13(b) illustrates that unless  $K$  is small, the basis functions become large near the ends of the approximation interval. This is problematic for controlling the pointwise error. We will see later that a better spacing of nodes can improve the situation substantially.

### 5.2.3 Taylor Series Expansion: Matching Derivatives

In the previous section, we used the Vandermonde system (5.4) to establish that matching  $K + 1$  sample values will uniquely specify a degree- $K$  least-squares poly-



**Figure 5.14:** Taylor series expansions of  $x_1(t)$  in (5.6) for degrees  $K = 1, 2, \dots, 5$ . Bold curve is the original function and light curves are labeled by the polynomial degree.

mial approximation to a real-valued function in  $\mathcal{L}^2([a, b])$ . Other ways of specifying  $K + 1$  constraints exist, one of the most familiar being the Taylor series expansion.

**Taylor Series Expansion** Assuming  $x$  has  $K$  derivatives at  $t_0$ , let

$$p_K(t) = \sum_{k=0}^K \frac{(t - t_0)^k}{k!} x^{(k)}(t_0). \quad (5.9)$$

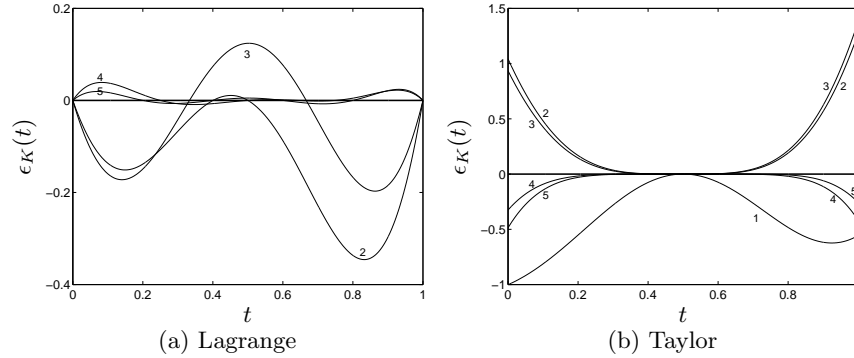
We have seen this expression in (P1.65-5); it is called a *Taylor series expansion around  $t_0$* . It has the property that  $p_K$  and  $x$  have equal derivatives of order  $0, 1, \dots, K$  at  $t_0$ . (The zeroth derivative is the function itself.)

**EXAMPLE 5.4 (TAYLOR SERIES EXPANSION)** Consider Taylor series expansions of the functions  $x_1$  and  $x_2$  in (5.6) around  $1/2$ . Both  $x_1$  and  $x_2$  are infinitely differentiable at  $1/2$ , so their Taylor series are easy to compute. Figure 5.14 shows  $x_1$  (bold) and its expansions of degree  $K = 1, 2, \dots, 5$ . The Taylor series for  $x_2$  is not interesting to plot because we have  $p_K(t) = t$  for all degrees  $\geq 1$ . While this is exact for  $t \in [0, 1/\sqrt{2}]$ , the error for  $t \in (1/\sqrt{2}, 1]$  cannot be reduced by increasing the degree.

A Taylor series expansion has the peculiar property of getting all its information around  $x$  from an infinitesimal interval around  $t_0$ . As Example 5.4 illustrated, this means the approximation can differ from the original function by an arbitrary amount if the function is discontinuous. For functions with sufficiently-many continuous derivatives, a precise error bound is given by the following theorem:

**THEOREM 5.2 (ERROR OF TAYLOR SERIES EXPANSION)** Let  $x$  have  $K + 1$  continuous derivatives on  $[a, b]$  for some  $K \geq 0$  and let  $t_0 \in [a, b]$ . Then, with  $p_K$  defined in (5.9),

$$\epsilon_K(t) = \frac{(t - t_0)^{K+1}}{(K + 1)!} x^{(K+1)}(\xi) \quad (5.10a)$$



**Figure 5.15:** Errors of polynomial interpolation of  $x_1(t) = t \sin 5t$  from Examples 5.3 and 5.4. Curves are labeled by the polynomial degree.

for some  $\xi$  between  $t$  and  $t_0$ . Thus,

$$|\epsilon_K(t)| \leq \frac{|t - t_0|^{K+1}}{(K+1)!} \max_{\xi \in [a, b]} |x^{(K+1)}(\xi)| \quad (5.10b)$$

for every  $t \in [a, b]$ .

The error bounds (5.7b) and (5.10b) are very similar. Both are proportional to

$$\frac{1}{(K+1)!} \max_{\xi \in [a, b]} |x^{(K+1)}(\xi)|,$$

the global smoothness measure. They differ in the dependence on  $t$ , but in a way that is consistent: If every  $t_k$  in (5.7b) is replaced by  $t_0$ , then the two error bounds are identical. Lagrange interpolation depends on the nodes being distinct so we cannot literally make all the  $t_k$ s equal. However, having  $K+1$  nodes distinct but closely clustered is similar to having derivatives up to order  $K$  at a single node. This is explored further in Exercise 5.5.

Moreover, these bounds require greater smoothness as the polynomial degree is increased. Furthermore, there exist infinitely-differentiable functions for which these bounds do not even decrease as  $K$  is increased; see Exercises 5.4 and 5.6.

### 5.2.4 Hermite Interpolation: Matching Points and Derivatives

A natural combination of the ideas of Lagrange interpolation and Taylor series is to determine a polynomial by fixing some number of derivatives at each of several nodes. This is called *Hermite interpolation*.

**EXAMPLE 5.5 (HERMITE INTERPOLATION)** Suppose that we are given the values of a function  $x$  and its derivative at  $t_0$  and  $t_1$ :

$$y_0 = x(t_0), \quad z_0 = x'(t_0), \quad y_1 = x(t_1), \quad z_1 = x'(t_1).$$

## 5.2. Approximation of Functions on Finite Intervals by Polynomials

471

We want to check that fixing these four values uniquely determines a cubic polynomial. We write the cubic polynomial and its derivative,

$$\begin{aligned} p_3(t) &= \alpha_0 + \alpha_1 t + \alpha_2 t^2 + \alpha_3 t^3, \\ p'_3(t) &= \alpha_1 + 2\alpha_2 t + 3\alpha_3 t^2, \end{aligned}$$

and find their values at  $t_0$  and  $t_1$ :

$$\begin{bmatrix} 1 & t_0 & t_0^2 & t_0^3 \\ 1 & t_1 & t_1^2 & t_1^3 \\ 0 & 1 & 2t_0 & 3t_0^2 \\ 0 & 1 & 2t_1 & 3t_1^2 \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ z_0 \\ z_1 \end{bmatrix}. \quad (5.11)$$

The matrix in this equation is invertible for any distinct pair  $(t_0, t_1)$  because its determinant is  $-(t_1 - t_0)^4 \neq 0$ . Thus, the polynomial is uniquely determined.

More generally, suppose that derivatives of order  $0, 1, \dots, d_k$  are specified at each distinct node  $t_k$  for  $k = 0, 1, \dots, L$ . Since the constraints are independent, a polynomial of degree  $K = (\sum_{k=0}^L (d_k + 1)) - 1$  can be uniquely determined. A proof of this and an error bound generalizing (5.7b) and (5.10b) are outlined in Exercise 5.7.

## 5.2.5 Minimax Polynomial Approximation

The techniques we have studied thus far determine a polynomial approximation through linear operations. Least-squares approximations—minimizing an  $\mathcal{L}^2$  error metric—come through the linear operation of orthogonal projection to a subspace, and the interpolation methods use some combination of samples of a function and its derivatives to fix a polynomial approximation through the solution of a system of linear equations. We now turn to the minimization of pointwise, or  $\mathcal{L}^\infty$ , error. Since  $\mathcal{L}^\infty([a, b])$  is not a Hilbert space, we do not have geometric notions such as the orthogonality of the error to the subspace of polynomials  $\mathcal{P}_K[a, b]$  to guide the determination of the optimal approximation. We will see that the optimal polynomial approximation is generally more difficult to compute, but interpolation with specially chosen nodes is nearly optimal.

Let  $x(t)$  be a real-valued function in  $\mathcal{L}^2([a, b])$ . An approximation  $\hat{x}$  that minimizes

$$\|x - \hat{x}\|_\infty = \max_{t \in [a, b]} |x(t) - \hat{x}(t)|$$

over some set of possibilities for  $\hat{x}$  is called a *minimax* approximation because it minimizes the maximum error. The following theorem shows that the set of polynomials is rich enough to enable arbitrarily-small  $\mathcal{L}^\infty$  error for any continuous function. A constructive proof is given in Exercise 5.8.

**THEOREM 5.3 (WEIERSTRASS APPROXIMATION THEOREM)** Let  $x$  be continuous on  $[a, b]$  and let  $\varepsilon > 0$ . Then, there exists a polynomial  $p$  for which

$$|\epsilon(t)| = |x(t) - p(t)| \leq \varepsilon \quad \text{for every } t \in [a, b]. \quad (5.12)$$

Denote by  $p_K(t)$  the minimax approximation among polynomials of degree at most  $K$ , and let  $\epsilon_{p,K}(t)$  be the error of that approximation with  $\mathcal{L}^\infty$  norm  $\|\epsilon_{p,K}\|_\infty$ . The theorem states that the mere continuity of  $x$  is enough to ensure that  $\|\epsilon_{p,K}\|_\infty$  can be made arbitrarily small by choosing  $K$  large enough. This contrasts starkly with the  $\mathcal{L}^\infty$  error bounds that we have seen thus far, (5.7b) for Lagrange interpolation and (5.10b) for Taylor series expansion, which require greater smoothness as the polynomial degree is increased.

Sometimes bounds can be unduly pessimistic even when performance is reasonable. With Lagrange interpolation, the difficulty from an  $\mathcal{L}^\infty$  perspective is that the maximum error between nodes can differ greatly, so the  $\mathcal{L}^\infty$  error can be large even when the function and its approximation are close over most of the approximation interval. When the nodes are evenly spaced, the maximum error tends to be largest near the ends of the approximation interval; we observed this in Figure 5.15(a), connected also to the large peaks shown in Figure 5.13. The large oscillations near the endpoints can be so dramatic that the  $\mathcal{L}^\infty$  error diverges as  $K$  increases, even for a  $C^\infty$  function; see Exercise 5.4. Thus, we must move beyond interpolation with evenly-spaced nodes for minimax approximation. Moreover, with an approximating polynomial  $q_K(t)$ , it never pays to have fewer than  $K+1$  points at which the error is zero. Both bounding the minimax error and computing a minimax approximation depend on understanding what happens between these points.

Let  $\{t_k\}_{k=0}^K \subset [a, b]$  be  $K+1$  distinct and ordered points selected to partition  $[a, b]$  into  $K+2$  subintervals:

$$[a, b] = \bigcup_{k=0}^{K+1} I_k \quad \text{where} \quad I_k = \begin{cases} [a, t_0], & \text{for } k=0; \\ [t_{k-1}, t_k], & \text{for } k=1, 2, \dots, K; \\ [t_K, b], & \text{for } k=K+1. \end{cases} \quad (5.13)$$

(While it is natural to think of  $\epsilon_{q,K}(t_k) = x(t_k) - q_K(t_k) = 0$  for  $k=0, 1, \dots, K$ , this is not necessary for the developments that follow.) Since the subintervals cover  $[a, b]$ , the  $\mathcal{L}^\infty$  error is the maximum of the errors on the subintervals:

$$\|\epsilon_{q,K}\|_\infty = \max_{k=0,1,\dots,K+1} \max_{t \in I_k} |\epsilon_{q,K}(t)|.$$

The following theorem shows that for a minimax approximation, the error  $\epsilon_{q,K}$  should oscillate in sign from one subinterval  $I_k$  to the next and the maximum absolute value of the difference should be the same on every subinterval.

**THEOREM 5.4 (DE LA VALLÉE-POUSSIN ALTERNATION THEOREM [6])** Let  $x$  be continuous on  $[a, b]$  and let  $[a, b]$  be partitioned as in (5.13) for some  $K \in \mathbb{N}$ . Suppose there exists a polynomial  $q_K$  of degree at most  $K$  and numbers  $s_k \in I_k$  for  $k=0, 1, \dots, K+1$ , such that  $\epsilon_{q,K}(s_k)$  alternates in sign. Then

$$\min_{k=0,1,\dots,K} |\epsilon_{q,K}(s_k)| \leq \|\epsilon_{p,K}\|_\infty \leq \|\epsilon_{q,K}\|_\infty, \quad (5.14)$$

where  $\epsilon_{p,K}$  is the minimax approximation error among polynomials of degree at most  $K$ .



## 5.2. Approximation of Functions on Finite Intervals by Polynomials

473

The first of the two inequalities in (5.14) is the interesting one; the second simply states that the minimax approximation  $p_K$  is at least as good as  $q_K$ .

To use the theorem, we must find an approximating polynomial  $q_K$  that creates enough alternations in the sign of the error. Then, the strongest statement is obtained by choosing  $s_k$  such that each  $|\epsilon_{q,K}(s_k)|$  is maximum on the subinterval  $I_k$ . The main result of the theorem is that there is no way to change the polynomial to push the error uniformly below the smallest of the local maxima of  $|\epsilon_{q,K}(t)|$ . Intuitively, pushing the worst (largest) of the local maxima down inevitably has the effect of pushing the best (smallest) of the local maxima up. The next theorem makes the stronger statement that equality of the local maxima happens if and only if the minimax approximation has been found.

**THEOREM 5.5 (CHEBYSHEV EQUIOSCILLATION THEOREM [6])** Let  $x$  be continuous on  $[a, b]$ , and let  $K \in \mathbb{N}$ . Denote the minimax approximation error among polynomials of degree at most  $K$  by  $\epsilon_{p,K}$ . The minimax approximation  $p_K$  is unique and determined by the following property: There are at least  $K + 2$  points

$$a \leq s_0 < s_1 < \cdots < s_{K+1} \leq b$$

for which

$$x(s_k) - p_K(s_k) = \sigma (-1)^k \|\epsilon_{p,K}\|_\infty, \quad k = 0, 1, \dots, K + 1,$$

where  $\sigma = \pm 1$ , independent of  $k$ .

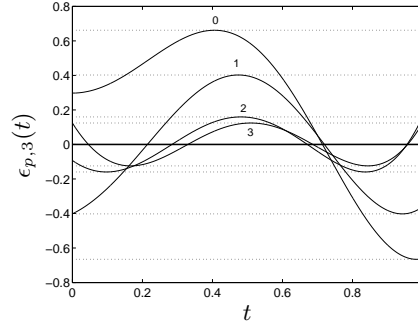
We illustrate both theorems by continuing our previous examples.

**EXAMPLE 5.6 (MINIMAX APPROXIMATION)** Consider again approximating  $x(t) = t \sin 5t$  on  $[0, 1]$ . To draw conclusions from Theorem 5.4, we must find a  $q_K(t)$  such that the error  $\epsilon_{q,K}(t)$  changes sign at least  $K + 1$  times. This is true for least-squares approximations computed in Example 5.2, shown in Figure 5.11. For each degree  $K \in \{0, 1, 2, 3\}$ , the error  $\epsilon_{q,K}(t)$  does indeed change sign at least  $K + 1$  times, allowing us to choose  $K + 2$  values  $s_k \in I_k$  to apply Theorem 5.4. (The error of the degree-2 approximation has one more sign change than necessary.) By choosing  $s_k$ s to give maxima of  $|\epsilon_{q,K}(t)|$  in each interval, we get from (5.14):

$$\begin{array}{ll} 0.459 & \leq \|\epsilon_{p,0}\|_\infty & 0.0717 & \leq \|\epsilon_{p,1}\|_\infty \\ 0.331 & \leq \|\epsilon_{p,1}\|_\infty & 0.0956 & \leq \|\epsilon_{p,2}\|_\infty \end{array}$$

The first four minimax polynomial approximations of  $x(t) = t \sin 5t$  and their  $\mathcal{L}^\infty$  errors are:

$$\begin{array}{ll} p_0(t) & \approx -0.297 & \|\epsilon_{p,0}\|_\infty & \approx 0.661 \\ p_1(t) & \approx 0.402 - 1.000t & \|\epsilon_{p,1}\|_\infty & \approx 0.402 \\ p_2(t) & \approx 0.0929 + 1.41t - 2.62t^2 & \|\epsilon_{p,2}\|_\infty & \approx 0.159 \\ p_3(t) & \approx -0.124 + 3.22t - 6.31t^2 + 2.13t^3 & \|\epsilon_{p,3}\|_\infty & \approx 0.124 \end{array}$$



**Figure 5.16:** Errors of minimax polynomial approximations  $p_K(t)$  of  $x(t) = t \sin 5t$  on  $[0, 1]$ , for degrees  $K = 0, 1, 2, 3$ . Curves are labeled by the polynomial degree. Dashed lines mark the maxima and minima, highlighting that for each degree, the minimum is the negative of the maximum.

These approximation errors  $\epsilon_{p,K}(t)$  are shown in Figure 5.16. The dotted lines highlight that the approximation error lies between  $\pm \|\epsilon_{p,K}\|_\infty$  and reaches these boundaries at least  $K + 2$  times, satisfying the condition of Theorem 5.5. The intuition behind the theorem is that to not reach the  $\pm \|\epsilon_{p,K}\|_\infty$  bounds  $K + 2$  times wastes some of the margin for error.

Computation of minimax approximations is generally very difficult because the values of the extrema of the error can depend on the polynomial coefficients in a complicated way. One common iterative algorithm is presented in Section 5.6.

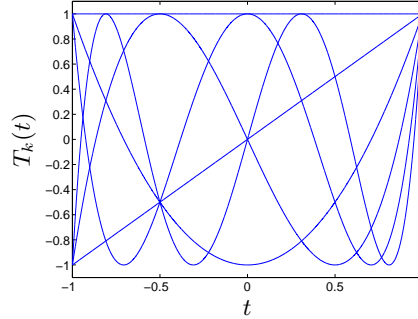
**Chebyshev Polynomials** An alternative to aiming for exact minimax approximations is to use an approximation that is simpler to compute but only nearly minimax. We now return to our observation regarding the weakness of the Lagrange interpolation with evenly-spaced nodes from an  $\mathcal{L}^\infty$  perspective; the error is large near the ends of the interval of approximation. One way to counter this is to have more nodes near the ends of the interval at the expense of having fewer near the center. We can estimate the improvement using the bound (5.7b). In fact, it seems sensible to minimize the factor  $\prod_{k=0}^K |t - t_k|$  from (5.7b), even though minimizing the bound is not the same as minimizing the  $\mathcal{L}^\infty$  error. When the interval of approximation is  $[-1, 1]$ , the resulting nodes are zeros of the Chebyshev polynomial of degree  $K$ . This result, discussed in more detail below, is one of many instances where we will encounter Chebyshev polynomials.

They are defined as

$$T_k(t) = \cos(k \arccos t), \quad k \in \mathbb{N}, \quad (5.15)$$

and are orthogonal on  $[-1, 1]$  with the weight function  $W(t) = (1 - t^2)^{-1/2}$ , that is,

$$\langle x, y \rangle = \int_{-1}^1 x(t)y(t) (1 - t^2)^{-1/2} dt. \quad (5.16)$$



**Figure 5.17:** Chebyshev polynomials up to degree 5,  $\{T_k\}_{k=0}^5$ .

The proof that these are indeed orthogonal polynomials is left for Solved Exercise 5.1. Chebyshev polynomials satisfy the recursion

$$T_{k+1}(t) = 2tT_k(t) - T_{k-1}(t), \quad (5.17)$$

with  $T_0(t) = 1$  and  $T_1(t) = t$ . The first few Chebyshev polynomials, plotted in Figure 5.17, are:

$$\begin{aligned} T_0(t) &= 1 & T_3(t) &= 4t^3 - 3t \\ T_1(t) &= t & T_4(t) &= 8t^4 - 8t^2 + 1 \\ T_2(t) &= 2t^2 - 1 & T_5(t) &= 16t^5 - 20t^3 + 5t \end{aligned}$$

The roots of  $T_k(t)$  are

$$t_m = \cos\left(\frac{2m+1}{2k}\pi\right), \quad m = 0, 1, \dots, k-1, \quad (5.18)$$

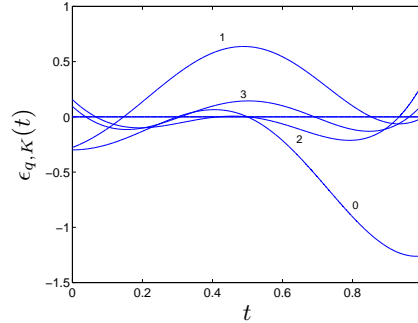
and the relative maxima and minima are

$$t_m = \cos\left(\frac{m}{k}\pi\right), \quad m = 0, 1, \dots, k. \quad (5.19)$$

Proofs of these expressions are left as Exercise 5.1.

From (5.15), it is clear that  $|T_k(t)| \leq 1$  for every  $k$  and  $t$ , see Figure 5.17. While the Legendre polynomials also satisfy this bound, they do not cover the interval  $[-1, 1]$  evenly; compare Figure 5.17 to Figure 5.10. Intuitively, the uniform magnitude of the Chebyshev polynomials makes the truncation error in approximating a continuous function with a linear combination of Chebyshev polynomials also have approximately uniform magnitude.

**Near Minimax Approximation** A related way to use Chebyshev polynomials is to interpolate with the roots (5.18) as nodes when approximating a function on  $[-1, 1]$ . Among all polynomials with degree  $K+1$  and leading coefficient 1, the scaled Chebyshev polynomial  $2^{-K}T_{K+1}(t)$  has minimum  $\mathcal{L}^\infty$  norm equal to  $2^{-K}$ .



**Figure 5.18:** Errors of near minimax polynomial approximations of  $x(t) = t \sin 5t$  on  $[0, 1]$ , for degrees  $K = 0, 1, 2, 3$ . Curves are labeled by the polynomial degree. Approximations are obtained by interpolation with Chebyshev nodes.

Therefore, the factor

$$\max_{t \in [-1, 1]} \prod_{k=0}^K |t - t_k|$$

in the maximum of the error bound (5.7b), is minimized by choosing  $\{t_k\}_{k=0}^K$  to be the  $K + 1$  zeros of  $T_{K+1}(t)$ . The bound then becomes

$$|\epsilon_{q,K}(t)| \leq \frac{1}{(K+1)! 2^K} \max_{\xi \in [-1, 1]} |x^{(K+1)}(\xi)| \quad (5.20)$$

for approximation of  $x(t)$  on  $[-1, 1]$  with a polynomial  $q_K$  of degree at most  $K$ . The error can be bounded relative to the minimax error as

$$\|\epsilon_{q,K}\|_{\infty} \leq \left( \frac{2}{\pi} \ln(K+1) + 2 \right) \|\epsilon_{p,K}\|_{\infty}, \quad (5.21)$$

so it is *near minimax* in a precise sense.

#### EXAMPLE 5.7 (NEAR MINIMAX APPROXIMATION WITH CHEBYSHEV POLYNOMIALS)

Return again to the approximation of  $x(t) = t \sin 5t$  on  $[0, 1]$ . If the interval of interest were  $[-1, 1]$ , we would obtain near minimax approximations satisfying (5.20) and (5.21) by interpolating with the roots of  $T_{K+1}(t)$  as the nodes. The only necessary modification is to map the roots from  $[-1, 1]$  to  $[0, 1]$  with an affine transformation. The errors of the resulting approximations are plotted for  $K \in \{0, 1, 2, 3\}$  in Figure 5.18.

Table 5.1 summarizes the  $\mathcal{L}^{\infty}$  error performances of various approximations from this and previous examples. The first four are significantly easier to compute than the minimax approximation. Least-squares approximation is a projection to the subspace of polynomials; it is optimal for  $\mathcal{L}^2$  error by definition, and its  $\mathcal{L}^{\infty}$  error is not necessarily small. A Taylor series expansion is accurate near the point at which the function is measured (assuming the function is smooth), but quite poor farther away. Interpolation using uniform nodes is improved upon by the

## 5.2. Approximation of Functions on Finite Intervals by Polynomials

477

Approximation method	Polynomial degree			
	0	1	2	3
Least-squares approximation	0.868	0.489	0.318	0.223
Lagrange interpolation	0.963	0.785	0.346	0.197
Taylor series expansion around 1/2	1.260	1.000	1.380	1.270
Minimax approximation	0.661	0.402	0.149	0.124
Near minimax approximation	1.260	0.637	0.298	0.144

**Table 5.1:** Summary of  $\mathcal{L}^\infty$  errors of approximations of  $x(t) = t \sin 5t$  on  $[0, 1]$ .

use of Chebyshev nodes; this becomes increasingly important as the polynomial degree is increased. Note also that (5.21) is satisfied.

**Filter Design** A common use of minimax approximation is in the design of FIR filters with linear phase. The design problem can be posed as one of finding coefficients of a polynomial so as to match a certain desired response, and the key is which criterion to optimize. Assume a filter with a symmetric impulse response of length  $L = 2K + 1$ , centered at the origin<sup>92</sup>

$$h_n = \begin{cases} h_{-n}, & \text{for } |n| \leq K; \\ 0, & \text{otherwise.} \end{cases} \quad (5.22)$$

Its frequency response is

$$H(e^{j\omega}) = \sum_{n=-K}^K h_n e^{-j\omega n} \stackrel{(a)}{=} h_0 + 2 \sum_{n=1}^K h_n \cos(n\omega), \quad (5.23)$$

where (a) follows from the desired linear phase (symmetry) of the filter and (2.275). We see that  $H(e^{j\omega})$  is a real and symmetric function of  $\omega$ . The goal now is to find the coefficients  $\{h_n\}_{n=0}^K$  so as to approximate a desired frequency response  $H^{(d)}(e^{j\omega})$ . For the sake of this discussion, we assume that the desired response corresponds also to a symmetric impulse response of real coefficients, or  $h_n^{(d)} = h_{-n}^{(d)}$  and  $h_n^{(d)} \in \mathbb{R}$ .

A first, obvious criterion is to use least-squares approximation, that is, minimize the quadratic error

$$\min_{\{h_n\}} \|H^{(d)}(e^{j\omega}) - H(e^{j\omega})\|_2^2 = \min_{\{h_n\}} \int_{-\pi}^{\pi} |H^{(d)}(e^{j\omega}) - H(e^{j\omega})|^2 d\omega. \quad (5.24)$$

By Parseval's equality (2.103), this is equivalent to minimizing the time-domain error

$$\min_{\{h_n\}} \|h_n^{(d)} - h_n\|_2^2 = \min_{\{h_n\}} \sum_{n=-\infty}^{\infty} (h_n^{(d)} - h_n)^2 = \min_{\{h_n\}} \sum_{n=-K}^K (h_n^{(d)} - h_n)^2. \quad (5.25)$$

<sup>92</sup>This will produce a noncausal filter, but one having a real frequency response. Once designed, one can make the filter causal by a right shift of  $K$ .

Since the cost is a sum of positive terms, this minimum is attained for

$$h_n = h_n^{(d)}, \quad \text{for } n = -K, -K+1, \dots, K. \quad (5.26)$$

In other words, the best least-squares approximation is simply the truncation of the desired filter's impulse response to its central  $2K+1$  terms. While the quadratic error is minimized, the maximum error can remain quite large. In particular, if the desired filter's frequency response is discontinuous, as, for example, in an ideal lowpass filter (2.107a), then the Gibbs phenomenon leads to oscillations that do not diminish in amplitude, whatever the length (see Figure 2.8).

Another approach is to use minimax approximation, that is, minimize the maximum error between  $H^{(d)}(e^{j\omega})$  and  $H(e^{j\omega})$ ,

$$\|\epsilon\|_\infty = |H^{(d)}(e^{j\omega}) - H(e^{j\omega})|_\infty = \max_{\omega \in (-\pi, \pi)} |H^{(d)}(e^{j\omega}) - H(e^{j\omega})|. \quad (5.27)$$

The goal is to minimize  $\epsilon_\infty$  over the possible choices of  $\{h_n\}_{n=0}^K$ . Such a minimax solution will satisfy the Chebyshev equioscillation theorem, Theorem 5.5, but for this, we need to turn our filter design problem into a polynomial approximation. To do so, note first that

$$T_n(\cos \omega) = \cos n\omega. \quad (5.28)$$

This allows us to turn (5.23) into a polynomial of  $\cos(\omega)$ . For example, if  $h = [d \ c \ b \ \boxed{a} \ b \ c \ d]$ , then (5.23) becomes

$$\begin{aligned} H(e^{j\omega}) &= a + 2b \cos \omega + 2c \cos 2\omega + 2d \cos 3\omega \\ &\stackrel{(a)}{=} a + 2b \cos \omega + 2c(2 \cos^2 \omega - 1) + 2d(4 \cos^3 \omega - 3 \cos \omega) \\ &= (a - 2c) + (2b - 6d) \cos \omega + 4c \cos^2 \omega + 8d \cos^3 \omega, \end{aligned}$$

where in (a) we used the expressions for the first few Chebyshev polynomials. The resulting filter's frequency response is a third-degree polynomial in  $\cos \omega$ . In general,

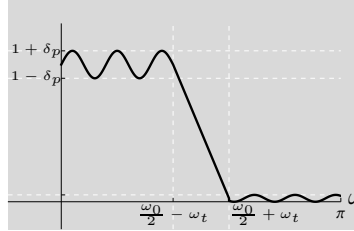
$$H(e^{j\omega}) = \sum_{k=0}^K c_k \cos^k \omega = P(t)|_{t=\cos \omega}, \quad (5.29)$$

and therefore, the filter design problem becomes a minimax polynomial approximation problem. In particular, it indicates a necessary condition for an optimal solution, namely that there will be  $K+2$  points with maximum error. This feature, called the equiripple property, is used in the algorithmic solution to the optimization problem, known as the Parks–McClellan algorithm.

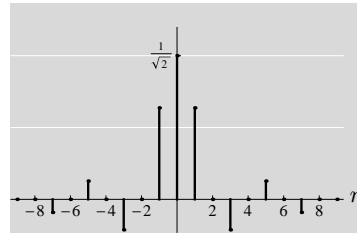
**EXAMPLE 5.8 (MINIMAX APPROXIMATION OF AN IDEAL LOWPASS FILTER)** We consider the design of an ideal lowpass filter, with a cut off frequency at  $\omega_0/2 = \pi/2$ ; its impulse response is given in (2.107b). Since an ideal filter cannot be attained with an FIR approximation, we will compare the best least-squares approximation with a minimax design. For the latter, we will allow a transition band between passband and stopband. Assume an FIR filter of length 15. The

## 5.2. Approximation of Functions on Finite Intervals by Polynomials

479



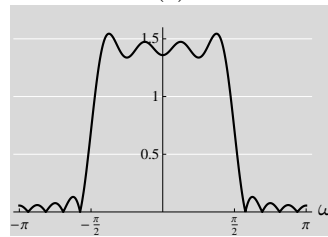
**Figure 5.19:** Specification for the approximation of a lowpass filter using minimax optimization. The transition band is of width  $2\omega_t$ , and  $\delta_p$ ,  $\delta_s$  are the error margins for passband and stopband, respectively.



(a)

FIGURE TBD

(b)



(c)

FIGURE TBD

(d)

**Figure 5.20:** Least-squares and minimax approximations to a halfband lowpass filter. (a) Impulse responses. (b) Difference. (c) Magnitude response. (d) Difference.

best least-squares approximation is the truncation of the sequence in (2.107b) to  $n = -7, -6, \dots, 7$ . Note that this is not the best least-squares approximation onto  $S = \text{BL}[-\pi/2, \pi/2]$  as in Theorem 4.8.

For a minimax approximation, introduce a transition band between  $\omega_0/2 - \omega_t$  and  $\omega_0/2 + \omega_t$ . This allows to smoothly move from the passband, where the response is close to 1, to the stopband, where it is close to 0, see Figure 5.19. Further specifying the approximation, we require the passband to be within  $1 \pm \delta_p$  and the stopband to be bounded by  $\delta_s$ , where  $\delta_p$  and  $\delta_s$  are imposed by the application domain of the filter (as is the width of the transition band  $2\omega_t$ ).

Of course, if  $\omega_t$ ,  $\delta_p$  and  $\delta_s$  are too stringent, a solutions might not exist. Choose, for example,

$$\omega_t = \frac{\pi}{10}, \quad \delta_p = 0.1, \quad \text{and} \quad \delta_s = 0.05. \quad (5.30)$$

The resulting filter (designed in Matlab using the *Remez* function) and the truncated ideal filter are shown in Figure 5.20(a), while in (b) the difference between them is magnified. The goal is to see how well the lowpass filter is approximated in frequency domain, shown in Figure 5.20. The different styles of approximation are clearly visible. While the minimax solution distributes the error evenly and transits smoothly from passband to stopband, the least-squares approximation hugs the specifications as close as possible until there is a large error, just at the transition.

To give some intuition why minimax designs are preferred over least-squares designs in signal processing applications, think of filtering of sine waves. In the minimax case, the variation of the filtering effect on two different frequencies is minimized, be it in the passband or the stopband. In the least-squares approximation, while most frequencies are treated close to the ideal case, the ones close to the transition suffer a large error. This, being at the boundary of the passband and stopband, can lead to noticeable errors, whereas the equitable distribution of errors in minimax is most likely unnoticeable.

### 5.3 Approximation of Functions by Splines

In the last section, we saw polynomial approximations that matched a function to be approximated either at specific points, or matched to its derivatives. This is typically done *globally* over an interval, sometimes leading to poor approximations (for example, at the boundaries). Two solutions can address this problem, namely minimax approximation, which we also saw in the last section, or local polynomial approximation using splines, which we introduce now. The idea of polynomial approximation using splines is to construct *shift-invariant subspaces* that span polynomials locally. Then, the smooth part of a function is well approximated by its projection onto the shift-invariant subspace. The prototype function used to generate the shift-invariant subspace must satisfy a simple condition to approximate polynomials, given by the Strang–Fix theorem, Theorem 5.7. The simplest case is given by B-splines, a generalization of the box function with many useful properties.

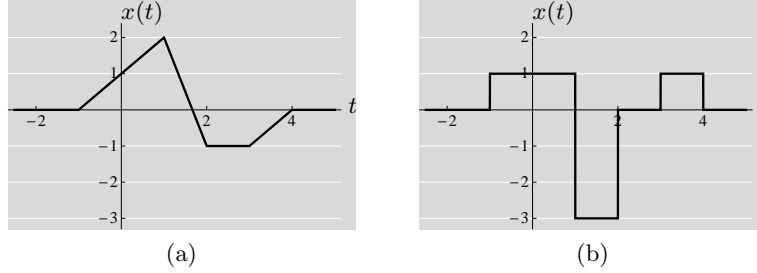
**B-Splines** Consider the elementary box function and its Fourier-transform pair, (3.76), with  $t_0 = 1$ . Call this the elementary B-spline of order 0, or a constant spline,

$$\beta^{(0)}(t) = \begin{cases} 1, & |t| \leq 1/2; \\ 0, & \text{otherwise,} \end{cases} \quad \xleftrightarrow{\text{FT}} \quad B^{(0)}(\omega) = \text{sinc}\left(\frac{\omega}{2}\right). \quad (5.31a)$$

Define the  $N$ th order B-spline by repeated convolution of  $\beta^{(0)}(t)$  with itself,

$$\beta^{(N)}(t) = \beta^{(N-1)}(t) * \beta^{(0)}(t), \quad \xleftrightarrow{\text{FT}} \quad B^{(N)}(\omega) = \left(\text{sinc}\left(\frac{\omega}{2}\right)\right)^{N+1}. \quad (5.31b)$$





**Figure 5.21:** Linear combination of linear splines  $\beta^{(1)}(t)$  showing that the result is continuous and differentiable almost everywhere. (a)  $x(t)$  with  $x_n = [\dots 0 \boxed{1} 2 - 1 - 10 \dots]$ . (b)  $x'(t)$ , which is well defined except at integers.

**Shift-Invariant Subspaces** Define the following shift-invariant subspaces:

$$S_N = \begin{cases} \text{span}(\{\beta^{(N)}(t-k)\}_{k \in \mathbb{Z}}), & N = 2k+1; \\ \text{span}(\{\beta^{(N)}(t-k-1/2)\}_{k \in \mathbb{Z}}), & N = 2k. \end{cases} \quad (5.32)$$

Functions belonging to these spaces, besides being invariant under integer shifts, have the very interesting property of having  $N-1$  continuous derivatives everywhere, and  $N$  continuous derivatives almost everywhere (except at integers). In other words, functions in  $S_N$  belong to  $C^{N-1}$  (and are almost  $C^N$ ); the proof is left for Exercise 5.11.

### 5.3.1 Approximation in Shift-Invariant Subspaces Using Splines

We now show how to use splines to approximate functions in shift-invariant subspaces. We start with a simple example.

**EXAMPLE 5.9 (LINEAR SPLINE BASIS)** Consider the first-order spline function  $\beta^{(1)}(t) = \beta^{(0)}(t) * \beta^{(0)}(t)$ . This is the hat function we have seen in (3.49a) and Figure 3.4(a); its Fourier transform was given in (3.49f) and Figure 3.4(b),

$$\beta^{(1)}(t) = \begin{cases} 1-|t|, & |t| < 1; \\ 0, & \text{otherwise.} \end{cases} \quad \xleftrightarrow{\text{FT}} \quad B^{(N)}(\omega) = \left( \text{sinc}\left(\frac{\omega}{2}\right) \right)^2. \quad (5.33)$$

The linear spline is continuous, and differentiable almost everywhere (except at  $-1, 0$ , and  $1$ ). Take any linear combination of integer-shifted linear spline,

$$x(t) = \sum_{k \in \mathbb{Z}} x_k \beta^{(1)}(t-k). \quad (5.34)$$

Figure 5.21 illustrates such a linear combination, where it can be seen that: (1)  $x(n) = x_n$ ; (2)  $x(t)$  is continuous; and (3)  $x(t)$  is differentiable, except at integers.

Given an arbitrary function  $x(t)$ , how can we compute its best approximation  $\hat{x}(t)$  in  $S_1$ ? Since the linear spline is not orthogonal to its integer shifts, we need to compute the dual basis, which is shift invariant too. That is, we

are looking for a function  $\tilde{\beta}^{(1)}(t)$  satisfying the classic biorthogonality relations (1.102),

$$\langle \tilde{\beta}^{(1)}(t), \beta^{(1)}(t-n) \rangle_t = \delta_n, \quad (5.35)$$

as well as

$$\tilde{\beta}^{(1)}(t) = \sum_{k \in \mathbb{Z}} c_k \beta^{(1)}(t-k). \quad (5.36)$$

Substituting (5.36) into (5.35), we need to find a sequence  $c_k$  such that

$$\sum_{k \in \mathbb{Z}} c_k \langle \beta^{(1)}(t-k), \beta^{(1)}(t-n) \rangle_t \stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} c_k a_{n-k}^{(1)} = \delta_n, \quad (5.37)$$

where in (a) we denote by  $a_n^{(1)}$  the deterministic autocorrelation function of  $\beta^{(1)}(t)$  evaluated at integers. This autocorrelation sequences is

$$a_{n-k}^{(1)} = \begin{cases} 2/3, & k = n; \\ 1/6, & k = n \pm 1; \\ 0, & \text{otherwise,} \end{cases} = \frac{2}{3}\delta_{n-k} + \frac{1}{6}\delta_{n-k+1} + \frac{1}{6}\delta_{n-k-1}, \quad (5.38)$$

because the linear splines overlap only when shifted by at most 1. With  $a^{(1)} = [\dots \ 0 \ 1/6 \ 2/3 \ 1/6 \ 0 \ \dots]$ , we can rewrite (5.37) as a convolution

$$\sum_{k \in \mathbb{Z}} c_k a_{n-k}^{(1)} = \delta_n \xrightarrow{\text{ZT}} C(z)A^{(1)}(z) = 1. \quad (5.39)$$

We invert  $A^{(1)}(z)$  to obtain  $C(z)$  as

$$\begin{aligned} C(z) &= \frac{1}{A^{(1)}(z)} = \frac{6}{4+z+z^{-1}} = -\frac{6z^{-1}}{(1-\alpha z^{-1})(1-(-1/\alpha)z^{-1})} \\ &= \frac{A}{1-\alpha z^{-1}} + \frac{B}{1-(-1/\alpha)z^{-1}} = -\frac{\sqrt{3}}{1-\alpha z^{-1}} + \frac{\sqrt{3}}{1-(-1/\alpha)z^{-1}}, \end{aligned}$$

with  $\alpha = \sqrt{3} - 2$  and where we used the partial fraction expansion method as explained in Section 2.5. For a stable sequence, we have two possibilities for the ROC:  $|z| < \alpha$ , leading to a right-sided sequence that cannot be an autocorrelation, and  $\alpha < |z| < 1/\alpha$ , leading to a symmetric two-sided autocorrelation sequence. We read off the individual terms from Table 2.6, to yield

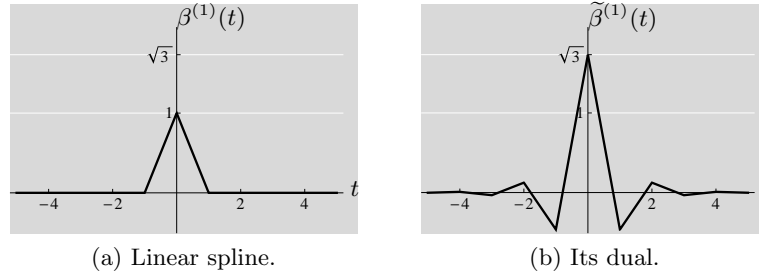
$$c_n = \sqrt{3}(\sqrt{3}-2)^{-n}u_{-n-1} + \sqrt{3}(\sqrt{3}-2)^n u_n = \sqrt{3}(\sqrt{3}-2)^{|n|}. \quad (5.40)$$

This was quite an involved a procedure; instead, a faster solution is the following. The impulse response  $c_n$  will be a symmetric geometric sequence,

$$c = [\dots \ \alpha^3 \ \alpha^2 \ \alpha \ \boxed{1} \ \alpha \ \alpha^2 \ \alpha^3 \ \dots].$$

Because of (5.39), its convolution with  $a_n^{(1)}$  has to equal 0, except at the origin,

$$\frac{1}{6} + \frac{2}{3}\alpha + \frac{1}{6}\alpha^2 = 0,$$



**Figure 5.22:** (a) The linear spline  $\beta^{(1)}(t)$  (hat function) (5.33) and (b) its dual  $\tilde{\beta}^{(1)}(t)$  using (5.40) and (5.36).

with roots  $\sqrt{3} \pm 2$ . By the same arguments as above, we choose ROC as  $2 - \sqrt{3} < |z| < 2 + \sqrt{3}$ , corresponding to a stable, symmetric, two-sided geometric sequence, as in (5.40). Using the coefficients we just found, we can compute  $\tilde{\beta}^{(1)}(t)$  as in (5.36), shown in Figure 5.22(b), together with the linear spline in Figure 5.22(a).

What we saw for the linear spline is generally true for a B-spline of any order; Exercise 5.12 shows it for the quadratic spline. Denote by  $a_n^{(N)}$  the deterministic autocorrelation function of  $\beta^{(N)}(t)$  evaluated at integers,

$$a_n^{(N)} = \langle \beta^{(N)}(t), \beta^{(N)}(t - n) \rangle_t. \quad (5.41a)$$

This sequence is nonzero for  $|n| \leq N$ , and we know it is symmetric and its DTFT is positive. According to (2.143b), its  $z$ -transform  $A^{(N)}(z)$  can be factored into

$$A^{(N)}(z) = R^{(N)}(z) R^{(N)}(z^{-1}), \quad (5.41b)$$

where  $R^{(N)}(z)$  has all zeros strictly inside the unit circle. Therefore, there exists an inverse  $C^{(N)}(z) = 1/A^{(N)}(z)$  such that its inverse  $z$ -transform,  $c_n^{(N)}$ , is a two-sided stable sequence. From this follows a biorthogonal dual function

$$\tilde{\beta}^{(N)}(t) = \sum_{k \in \mathbb{Z}} c_k^{(N)} \beta^{(N)}(t - k), \quad (5.41c)$$

satisfying, by construction,

$$\langle \tilde{\beta}^{(N)}(t), \beta^{(N)}(t - n) \rangle_t = \delta_n. \quad (5.41d)$$

This allows us to approximate an arbitrary function  $x(t)$  by its orthogonal projection onto the space of B-splines of order  $N$ ,

$$\hat{x}(t) = \sum_{k \in \mathbb{Z}} \langle x(t), \tilde{\beta}^{(N)}(t - k) \rangle_t \beta^{(N)}(t - k). \quad (5.41e)$$

### 5.3.2 Approximation in Shift-Invariant Subspaces Using Orthogonalized Splines

Synthesizing functions based on B-splines is very convenient, since the basis functions  $\beta^{(N)}(t - n)$  are of finite support. However, the dual functions  $\tilde{\beta}^{(N)}(t)$  are

of infinite support, albeit with fast, exponential decay. An alternative is to derive orthonormal bases for the spaces  $S_N$ . This can be done via an orthogonalization procedure of the spline basis  $\{\beta^{(N)}(t-n)\}_{n \in \mathbb{Z}}$ . We use linear splines to illustrate the method.

EXAMPLE 5.10 (ORTHOGONALIZED LINEAR SPLINE BASIS) We want to derive a shift-invariant basis  $\{\varphi^{(1)}(t-n)\}_{n \in \mathbb{Z}}$  for  $S_1$ , or

$$\text{span} \left( \{\varphi^{(1)}(t-n)\}_{n \in \mathbb{Z}} \right) = \text{span} \left( \{\beta^{(1)}(t-n)\}_{n \in \mathbb{Z}} \right), \quad (5.42a)$$

such that it is orthonormal,

$$\langle \varphi^{(1)}(t), \varphi^{(1)}(t-n) \rangle_t = \delta_n.$$

Because  $\{\beta^{(1)}(t-n)\}_{n \in \mathbb{Z}}$  form a basis for  $S_1$ , we can express our desired basis functions as a linear combination of  $\beta^{(1)}(t-n)$ ,

$$\varphi^{(1)}(t) = \sum_{k \in \mathbb{Z}} d_k \beta^{(1)}(t-k). \quad (5.42b)$$

Using the orthonormality of basis functions,

$$\begin{aligned} \langle \varphi^{(1)}(t), \varphi^{(1)}(t-n) \rangle_t &= \sum_{k \in \mathbb{Z}} \sum_{\ell \in \mathbb{Z}} d_k d_\ell \langle \beta^{(1)}(t-k), \beta^{(1)}(t-n-\ell) \rangle_t \\ &\stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} \sum_{\ell \in \mathbb{Z}} d_k d_\ell a_{n+\ell-k}^{(1)} \\ &\stackrel{(b)}{=} \sum_{k \in \mathbb{Z}} \sum_{k' \in \mathbb{Z}} d_k d_{k+k'} a_{n-k'}^{(1)} \\ &\stackrel{(c)}{=} \sum_{k' \in \mathbb{Z}} a_{k'} a_{n-k'}^{(1)} \stackrel{(d)}{=} a * a^{(1)}, \end{aligned} \quad (5.43)$$

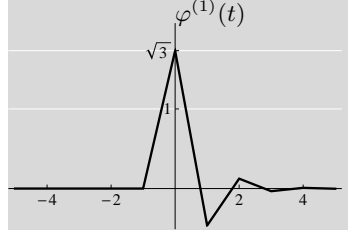
where (a) follows from (5.38) and the symmetry of  $a^{(1)}$ ; (b) from the change of variable  $k' = k - \ell$  and  $d_k d_{k-k'} = d_k d_{k+k'}$ ; in (c) we named the autocorrelation  $a_{k'} = \sum_{k'} d_k d_{k+k'}$ ; and (d) from the fact it is a convolution. In  $z$ -transform domain, orthonormality becomes

$$A(z)A^{(1)}(z) = 1, \quad (5.44)$$

where  $A(z)$  is the  $z$ -transform of the deterministic autocorrelation of  $d_k$ .

Of course, we know that the solution to (5.44) is going to be the same as the solution to (5.39). We now proceed to find it using this method. We need to take the spectral root of  $A(z)$  via  $A(z) = D(z)D(z^{-1})$ . A possible sequence that has (5.40) as an autocorrelation is

$$d_n = \begin{cases} \sqrt{3}(\sqrt{3}-2)^n, & \text{for } n \geq 0; \\ 0, & \text{otherwise,} \end{cases} \quad (5.45)$$



**Figure 5.23:** Orthonormal basis function  $\varphi^{(1)}(t)$ , which, together with its integer shifts, spans  $S_1$ , the same space spanned by the linear spline and its integer shifts.

(see also Exercise 2.4). The resulting orthonormal basis function,

$$\varphi^{(1)}(t) = \sum_{k=0}^{\infty} d_k \beta^{(1)}(t - k), \quad (5.46)$$

is shown in Figure 5.23.

As for the biorthogonal case of the previous subsection, the orthogonalization method generalizes to any B-spline. In (5.41b), take the spectral root with all zeros inside the unit circle. Its inverse is stable, and leads to a right-sided sequence  $d_n^{(N)}$  from which one derives

$$\varphi^{(N)}(t) = \sum_{k=0}^{\infty} d_k^{(N)} \beta^{(N)}(t - k). \quad (5.47)$$

This function and its integer shifts span  $S_N$ . Any function  $x(t)$  can be approximated by

$$\hat{x}(t) = \sum_{k \in \mathbb{Z}} \langle x(t), \varphi^{(N)}(t - k) \rangle_t \varphi^{(N)}(t - k). \quad (5.48)$$

One remarkable property we saw both for  $\beta^{(0)}(t)$  and  $\beta^{(1)}(t)$  is their interpolation property, that is,  $\beta^{(0)}(n) = \delta_n$  and  $\beta^{(1)}(n) = \delta_n$ . However, B-splines of higher order do not have this property; for this, one needs to use *cardinal splines* instead. Exercise 5.13 shows a general orthogonalization procedure.

### 5.3.3 Continuous-Time Processing Using Discrete-Time Operators in Spline Spaces

From our discussion so far, it is clear that there is a very tight bond between the sequence  $x_n$  and the function  $x(t)$  generated using a B-spline and its shifts,

$$x(t) = \sum_{k \in \mathbb{Z}} x_k \beta^{(N)}(t - k). \quad (5.49)$$

An interesting manifestation of that bond is that one can perform continuous-time processing on  $x(t)$  by performing discrete-time processing on  $x_n$ . The idea is intuitive, since, if a signal is in  $S_N$ , its derivative will be in  $S_{N-1}$ , and conversely, its

integral will be in  $S_{N+1}$ . Throughout, we will be manipulating Dirac delta functions; as before, we proceed formally without worrying about technical issues as these were discussed in Chapter 3.

**Computing Derivatives** To compute a derivative of  $x(t)$ , we need to compute a derivative of the splines used in the expansion (see (5.49)). We do that by exploiting the fact that they are formed as successive convolutions of box functions (5.31b) and using the derivative formula for convolution (3.66c).

We start with the causal version of the constant spline

$$\beta^{(0)}(t) = \begin{cases} 1, & \text{for } 0 \leq t < 1; \\ 0, & \text{otherwise,} \end{cases} = u(t) - u(t-1), \quad (5.50a)$$

where  $u(t)$  is the Heaviside function (3.8). The derivative of the constant spline is<sup>93</sup>

$$\Delta(t) = \frac{d}{dt}\beta^{(0)}(t) = \frac{d}{dt}(u(t) - u(t-1)) \stackrel{(a)}{=} \delta(t) - \delta(t-1), \quad (5.50b)$$

where (a) follows from (3.9). Then, the derivative of the  $N$ th-order spline is

$$\frac{d}{dt}(\beta^{(N)}(t)) \stackrel{(a)}{=} \frac{d}{dt}(\beta^{(N-1)}(t) * \beta^{(0)}(t)) = \beta^{(N-1)}(t) * \Delta(t), \quad (5.51)$$

where (a) follows from (5.31b).

We now rewrite  $x(t)$  starting from (5.49) as

$$\begin{aligned} x(t) &= \sum_{k \in \mathbb{Z}} x_k \beta^{(N)}(t-k) \stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} x_k \int_{-\infty}^{\infty} \delta(\tau-k) \beta^{(N)}(t-\tau) d\tau \\ &\stackrel{(b)}{=} \int_{-\infty}^{\infty} \sum_{k \in \mathbb{Z}} x_k \delta(\tau-k) \beta^{(N)}(t-\tau) d\tau \stackrel{(c)}{=} \int_{-\infty}^{\infty} \tilde{x}(\tau) \beta^{(N)}(t-\tau) d\tau \\ &\stackrel{(d)}{=} \tilde{x}(t) * \beta^{(N)}(t) \end{aligned} \quad (5.52)$$

where (a) follows from the sifting property of the Dirac delta function in Table 3.1; in (b) we exchanged the order of integration and summation; in (c) we named  $\tilde{x}(\tau) = \sum_{k \in \mathbb{Z}} x_k \delta(\tau-k)$ ; and (d) follows from the convolution formula (3.36).

To find the derivative of  $x(t)$ , write

$$\begin{aligned} \frac{d}{dt}x(t) &\stackrel{(a)}{=} \frac{d}{dt}(\tilde{x}(t) * \beta^{(N)}(t)) \stackrel{(b)}{=} \tilde{x}(t) * \frac{d}{dt}\beta^{(N)}(t) \\ &\stackrel{(c)}{=} \tilde{x}(t) * \Delta(t) * \beta^{(N-1)}(t), \end{aligned} \quad (5.53)$$

<sup>93</sup>This derivative is in the sense of distributions.

## 5.3. Approximation of Functions by Splines

487

where (a) follows from (5.52); (b) from (3.66c); and (c) from (5.51). Now

$$\begin{aligned}
 \tilde{x}(t) * \Delta(t) &\stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} x_k \delta(t - k) * (\delta(t) - \delta(t - 1)) \\
 &\stackrel{(b)}{=} \sum_{k \in \mathbb{Z}} x_k \delta(t - k) - \sum_{k \in \mathbb{Z}} x_k \delta(t - k - 1) \\
 &\stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} (x_k - x_{k-1}) \delta(t - k), \tag{5.54}
 \end{aligned}$$

where (a) uses (5.50b); (b) uses the convolution property of the Dirac delta function from Table 3.1; and (c) gathers terms using a change of variable.

Starting from (5.53), compute the derivative of a function  $x(t)$  in  $S_N$  as

$$\begin{aligned}
 \frac{d}{dt}x(t) &= \tilde{x}(t) * \Delta(t) * \beta^{(N-1)}(t) \\
 &\stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} (x_k - x_{k-1}) \delta(t - k) * \beta^{(N-1)}(t) \\
 &\stackrel{(b)}{=} \sum_{k \in \mathbb{Z}} x'_k \delta(t - k) * \beta^{(N-1)}(t) \stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} x'_k \beta^{(N-1)}(t - k), \tag{5.55}
 \end{aligned}$$

where (a) follows from (5.54); in (b) we denoted by  $x'$  the first-order difference of the sequence  $x$ , the *discrete derivative*; and (c) follows from the sifting property of the Dirac delta function from Table 3.1.

In other words, to compute a continuous-time derivative of a function  $x(t) \in S_N$ , we can use its discrete-time derivative sequence  $x'_n$  and convolve it with splines; the resulting derivative  $x'(t)$  belongs to  $S_{N-1}$ ,

$$x(t) = \sum_{k \in \mathbb{Z}} x_k \beta^{(N)}(t - k), \tag{5.56a}$$

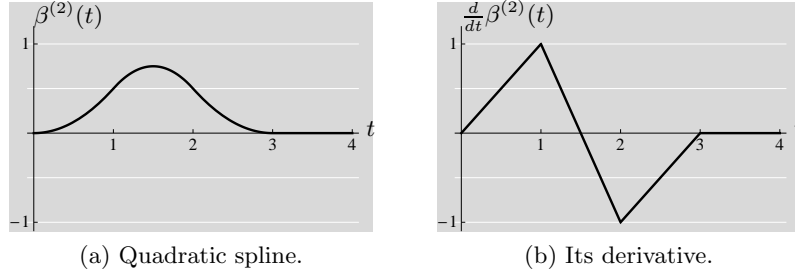
$$\frac{d}{dt}x(t) = \sum_{k \in \mathbb{Z}} x'_k \beta^{(N+1)}(t - k). \tag{5.56b}$$

**EXAMPLE 5.11 (DIFFERENTIATION IN A SPLINE BASIS)** We compute derivatives in  $S_2$ , spanned by  $\{\beta^{(2)}(t - k)\}_{k \in \mathbb{Z}}$ , where  $\beta^{(2)}(t)$  is in its causal version, supported on  $[0, 3]$  (we computed it as convolution of the constant and linear splines),

$$\beta^{(2)}(t) = \begin{cases} t^2/2, & \text{for } 0 \leq t < 1; \\ -t^2 + 3t - 3/2, & \text{for } 1 \leq t < 2; \\ t^2/2 - 3t + 9/2, & \text{for } 2 \leq t < 3; \\ 0, & \text{otherwise.} \end{cases} \tag{5.57}$$

Its derivative is continuous and piecewise linear,

$$\frac{d}{dt}\beta^{(2)}(t) = \begin{cases} t, & \text{for } 0 \leq t < 1; \\ -2t + 3, & \text{for } 1 \leq t < 2; \\ t - 3, & \text{for } 2 \leq t < 3; \\ 0, & \text{otherwise.} \end{cases}$$



**Figure 5.24:** Computing continuous-time derivatives using discrete-time ones.

The quadratic spline and its derivative are shown in Figure 5.24(a) and (b), respectively. The above is equal to

$$\beta^{(1)}(t) - \beta^{(1)}(t-1) = \Delta(t) * \beta^{(1)}(t),$$

as shown in Figure 5.24.

To compute a derivative of a function in  $S_N$ , we turn back to Example 5.9 and Figure 5.21. The signal  $x(t)$  is generated from the sequence  $x$ ,

$$\begin{aligned} x &= [\dots, 0, \boxed{1}, 2, -1, -1, 0, 0, \dots], \\ x' &= [\dots, 0, \boxed{1}, 1, -3, 0, 1, 0, \dots]. \end{aligned}$$

According to (5.55), it is easy to see that we obtain  $x'(t)$  as in Figure 5.21(b) (note that there is a left-shift by 1, since in Example 5.9, the hat function is centered at the origin rather than being causal).

**Computing Integrals** We have just seen how to compute continuous-time derivatives using discrete-time ones, with splines of lower order. It is tempting to do the reverse, which is indeed possible with some care. Let us look at (5.51). The primitive of the right side is equal to  $\beta^{(N)}(t)$ , up to a constant, or

$$\beta^{(N)}(t) = \int_{-\infty}^t \Delta(\tau) * \beta^{(N-1)}(\tau) d\tau. \quad (5.58)$$



## 5.3. Approximation of Functions by Splines

489

We now compute the integral of a function  $x(t)$  in  $S_N$ ,

$$\begin{aligned}
 \int_{-\infty}^t x(\tau) d\tau &\stackrel{(a)}{=} \int_{-\infty}^t \sum_{k \in \mathbb{Z}} x_k \beta^{(N)}(\tau - k) d\tau \stackrel{(b)}{=} \sum_{k \in \mathbb{Z}} \int_{-\infty}^t x_k \beta^{(N)}(\tau - k) d\tau \\
 &\stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} \int_{-\infty}^t \left( \sum_{n=-\infty}^k x_n - \sum_{n=-\infty}^{k-1} x_n \right) \beta^{(N)}(\tau - k) d\tau \\
 &\stackrel{(d)}{=} \sum_{k \in \mathbb{Z}} \int_{-\infty}^t (X_k - X_{k-1}) \beta^{(N)}(\tau - k) d\tau \\
 &\stackrel{(e)}{=} \sum_{k \in \mathbb{Z}} X_k \int_{-\infty}^t (\beta^{(N)}(\tau - k) - \beta^{(N)}(\tau - k + 1)) d\tau \\
 &\stackrel{(f)}{=} \sum_{k \in \mathbb{Z}} X_k \beta^{(N+1)}(t - k), \tag{5.59}
 \end{aligned}$$

where (a) follows from the assumption that  $x(t) \in S_N$ ; in (b) we exchanged order of summation and integration; in (c) we expressed  $x_k$  as the difference of two sums; in (d) we named those sums *discrete integrals*  $X_n = \sum_{\ell=-\infty}^n x_\ell$  of the sequence  $x$ ; (e) follows from change of variable  $k' = k - 1$ ; and (f) from (5.58).

In other words, to compute a continuous-time integral of a function  $x(t) \in S_N$ , we can use its discrete-time integral sequence  $X_k$  as weighting coefficients for the splines; the resulting integral belongs to  $S_{N+1}$ ,

$$x(t) = \sum_{k \in \mathbb{Z}} x_k \beta^{(N)}(t - k), \tag{5.60a}$$

$$\int_{-\infty}^t x(\tau) d\tau = \sum_{k \in \mathbb{Z}} X_k \beta^{(N+1)}(t - k). \tag{5.60b}$$

**EXAMPLE 5.12 (INTEGRATION IN A SPLINE BASIS)** Consider a function  $x(t)$  in  $S_0$ . According to (5.59),

$$\int_{-\infty}^t x(\tau) d\tau = \sum_{k \in \mathbb{Z}} X_k \beta^{(1)}(t - k).$$

For simplicity, assume that  $x(t) = 0$  for  $t < 0$  and  $t > L$  and  $\sum_{n=0}^{L-1} x_n = 0$ , which also means  $\int_{-\infty}^{\infty} x(t) dt = 0$ . The example shown in Figure 5.21(b) satisfies this requirement, where the sequence and its primitive are (the derivative and the sequence in the figure)

$$\begin{aligned}
 x &= [\dots \ 0 \ \boxed{1} \ 1 \ -3 \ 0 \ 1 \ 0 \ \dots], \\
 X &= [\dots \ 0 \ \boxed{1} \ 2 \ -1 \ -1 \ 0 \ 0 \ \dots].
 \end{aligned}$$

As was seen in Example 5.9,

$$x(t) = \sum_{k \in \mathbb{Z}} x_k \beta^{(0)}(t - k) \quad \text{and} \quad (5.61a)$$

$$\int_{-\infty}^t x(\tau) d\tau = \sum_{k \in \mathbb{Z}} X_k \beta^{(1)}(t - k) \quad (5.61b)$$

are related by integration, verifying (5.59).

Because of their simple form as piecewise polynomials, B-splines have a number of remarkable properties, as we have just seen. For example, another interesting result involving splines is that inner products between an arbitrary function  $x(t)$  and a spline  $\beta^{(N)}(t)$  can be computed by integrating  $x(t)$   $N + 1$  times, evaluating this integral at integers, and then taking  $N + 1$  discrete differences of the resulting sequence; see Exercise 5.14. B-splines are the simplest members of the spline family. Many generalizations are possible (for example, nonuniform, exponential); for pointers.

### 5.3.4 Polynomial Reproduction and Strang-Fix Theorem

The span of B-splines and their integer shifts was introduced in (5.32) as spaces  $S_N$ . We now show that polynomials of degree  $N$  belong to these spaces, that is,

$$p_N(t) = \sum_{k \in \mathbb{Z}} \alpha_k \beta^{(N)}(t - k). \quad (5.62)$$

This is a particular case of a more general result, the Strang-Fix theorem, to be discussed shortly. We start with an example.

**EXAMPLE 5.13 (POLYNOMIAL REPRODUCTION WITH THE QUADRATIC SPLINE)**  
We consider the quadratic B-spline as in (5.57), except centered.

- (i) We first show that  $\{\beta^{(2)}(t - k)\}_{k \in \mathbb{Z}}$  reproduce constants, that is,

$$\sum_{n \in \mathbb{Z}} \beta^{(2)}(t - n) = 1. \quad (5.63a)$$

The sum is a periodic function of period 1, and only 3 copies of  $\beta^{(2)}$  overlap at any point. It is enough to prove that  $\beta^{(2)}(t + 1) + \beta^{(2)}(t) + \beta^{(2)}(t - 1) = 1$  on the interval  $[-1/2, 1/2]$ . Apart from the central term, the left tail is right shifted by 1, and the right tail is left shifted by 1. Putting the three components together,

$$\frac{1}{2}(t - \frac{1}{2})^2 + \frac{3}{4} - t^2 + \frac{1}{2}(\frac{1}{2} - t)^2 = \frac{1}{4} - t + t^2 + \frac{3}{4} - t^2 = 1.$$

- (ii) We now show  $\{\beta^{(2)}(t - k)\}_{k \in \mathbb{Z}}$  reproduce linear terms, that is,

$$\sum_{n \in \mathbb{Z}} n \beta^{(2)}(t - n) = t. \quad (5.63b)$$

## 5.3. Approximation of Functions by Splines

491

Consider the interval  $[n - 1/2, n + 1/2]$ . The overlapping left, central and right portions of  $\beta^{(2)}$  are

$$\frac{1}{2}(n + \frac{1}{2} - t)^2, \quad \frac{3}{4} - (t - n)^2, \quad \text{and} \quad \frac{1}{2}(t - n + \frac{1}{2})^2,$$

and they are weighted by  $n - 1$ ,  $n$  and  $n + 1$ , respectively,

$$\frac{1}{2}(n - 1)(n + \frac{1}{2} - t)^2 + n(\frac{3}{4} - (t - n)^2) + \frac{1}{2}(n + 1)(t - n + \frac{1}{2})^2 = t.$$

(iii) Finally, we show  $\{\beta^{(2)}(t - k)\}_{k \in \mathbb{Z}}$  reproduce quadratic terms, that is,

$$\sum_{n \in \mathbb{Z}} n^2 \beta^{(2)}(t - n) = t^2 + \text{constant}. \quad (5.63c)$$

We can check that

$$\frac{1}{2}(n - 1)^2(n + \frac{1}{2} - t)^2 + n^2(\frac{3}{4} - (t - n)^2) + \frac{1}{2}(n + 1)^2(t + \frac{1}{2} - n)^2 = t^2 + \frac{1}{4}.$$

Then, any polynomial of second degree can be written as a linear combination,

$$a_2 t^2 + a_1 t + a_0 = \sum_{n \in \mathbb{Z}} \alpha_n \beta^{(2)}(t - n). \quad (5.64)$$

To prove polynomial reproduction by B-splines, we can use the derivative formula (5.55) and the integral formula (5.59). We claim that

$$\sum_n n^k \beta^{(N)}(t - n) = \sum_{\ell=0}^k a_\ell t^\ell, \quad k = 0, 1, \dots, N. \quad (5.65)$$

Using the derivative formula  $k$  times shows that the linear combination is at most a polynomial of degree  $k$ , while using the integral formula  $k$  times shows that the result is a polynomial of degree  $k$  (see Exercise 5.15). Implicitly, we use the following result:

**PROPOSITION 5.6 (REPRODUCTION OF IDENTITY)** Let  $\varphi(t) \in \mathcal{L}^1(\mathbb{R})$  and let  $\varphi_1(t)$  be its periodized version with period 1 as in (3.40). Then, the following form a Fourier-transform pair:

$$\varphi_1(t) = \sum_{n \in \mathbb{Z}} \varphi(t - n) = 1 \quad \xleftrightarrow{\text{FT}} \quad \Phi(2\pi k) = \delta_k. \quad (5.66)$$

*Proof.* Since  $\varphi_1(t)$  is periodic, it can be represented as a Fourier series (3.90b),

$$\varphi_1(t) = \sum_{k \in \mathbb{Z}} \Phi_{1,k} e^{j(2\pi/T)kt},$$

with coefficients from (3.90a)

$$\begin{aligned}
 \Phi_{1,k} &= \int_{-1/2}^{1/2} \varphi_1(t) e^{-j(2\pi/T)kt} dt \stackrel{(a)}{=} \int_{-1/2}^{1/2} \left( \sum_{n \in \mathbb{Z}} \varphi(t-n) \right) e^{-j(2\pi/T)kt} dt \\
 &\stackrel{(b)}{=} \sum_{n \in \mathbb{Z}} \int_{-1/2}^{1/2} \varphi(t-n) e^{-j(2\pi/T)kt} dt \stackrel{(c)}{=} \sum_{n \in \mathbb{Z}} \int_{-1/2-n}^{1/2-n} \varphi(\tau) e^{-j2\pi k(\tau+n)} d\tau \\
 &\stackrel{(d)}{=} \int_{-\infty}^{\infty} \varphi(\tau) e^{-j2\pi k\tau} d\tau \stackrel{(e)}{=} \Phi(2\pi k),
 \end{aligned}$$

where (a) follows from the definition of  $\varphi_1$  in (5.66); in (b) we interchanged the sum and integral; (c) follows from the change of variable  $\tau = t - n$ ; (d) from periodicity of  $e^{-j2\pi k\tau}$  and we also merged all individual unit-length intervals into a single interval over  $\mathbb{R}$ ; and (e) is the Fourier transform evaluated at  $\omega = 2\pi k$ . If  $\varphi_1(t) = 1$ , then  $\Phi_{1,k} = \Phi(2\pi k) = \delta_k$ ; conversely, if  $\Phi(2\pi k) = \Phi_{1,k} = \delta_k$  then  $\varphi_1(t) = 1$ .

An immediate consequence of this proposition is that for B-splines,

$$\sum_{n \in \mathbb{Z}} \beta^{(N)}(t-n) = 1, \quad (5.67)$$

which follows from (5.31b). A generalization of the above result is the following theorem:

**THEOREM 5.7 (POLYNOMIAL REPRODUCTION (STRANG–FIX))** Consider a function  $\varphi(t)$  and an integer  $K$ . If  $\varphi(t)$  has sufficient localization,

$$\int_{-\infty}^{\infty} (1 + |t|^K) |\varphi(t)| dt < \infty, \quad (5.68)$$

then the following are equivalent:

- (i) Polynomials of degree  $k \leq K$  are in the span of  $\{\varphi(t-n)\}_{n \in \mathbb{Z}}$ .
- (ii) The Fourier transform of  $\varphi(t)$ ,  $\Phi(\omega)$  and its  $K$  derivatives satisfy

$$\Phi^{(k)}(2\pi\ell) = \delta_k \delta_\ell, \quad k = 0, 1, \dots, K, \quad \ell \in \mathbb{Z}. \quad (5.69)$$

The proof of the theorem can be found in the book of Strang and Fix, see *Further Reading*. Solved Exercise 5.2 proves the theorem when  $\varphi(t)$  is an interpolating function, that is, when

$$\varphi(n) = \delta_n, \quad n \in \mathbb{Z}. \quad (5.70)$$

In the statement of the theorem, the localization property (5.68) is trivially satisfied by any  $\varphi(t)$  having finite support; for infinitely-supported  $\varphi(t)$ , however, sufficient decay is required. B-splines of order  $N$  have exactly  $N$ th-order zeros at nonzero multiples of  $2\pi$  since they are the product of  $N$  sinc functions (see (5.31b)). Therefore, they reproduce polynomials of degree up to  $N$ , as we had seen earlier.

EXAMPLE 5.14 (POLYNOMIAL REPRODUCTION) The quadratic B-spline has Fourier transform (from (5.31b))

$$B^{(2)}(\omega) = \left( \text{sinc}\left(\frac{\omega}{2}\right) \right)^3.$$

Let us check the second condition in the theorem, (5.69), for  $k = 0, 1, 2$ . For  $k = 0$ , we simply have the sinc function to the third power, so clearly

$$B^{(2)}(2\pi\ell) = (\text{sinc}(\pi\ell))^3 = \delta_\ell.$$

For the first derivative  $k = 1$ , we get

$$\begin{aligned} \left. \frac{d}{d\omega} \left( B^{(2)}(\omega) \right) \right|_{2\pi\ell} &= \left. \frac{d}{d\omega} \left( \text{sinc}\left(\frac{\omega}{2}\right)^3 \right) \right|_{2\pi\ell} = \frac{3}{2} \left( \text{sinc}\left(\frac{\omega}{2}\right) \right)^2 \left. \frac{d}{d\omega} \left( \text{sinc}\left(\frac{\omega}{2}\right) \right) \right|_{2\pi\ell} \\ &= (\text{sinc}(\pi\ell))^2 \left. \frac{d}{d\omega} (\text{sinc}(\pi\ell)) \right|_{2\pi\ell} = 0, \end{aligned} \quad (5.71)$$

because  $\text{sinc}(\omega/2)$  is symmetric around 0, and thus its derivative is 0 at the origin where  $\text{sinc}(\pi\ell)$  is nonzero. Taking one more derivative of (5.71) leads to

$$\begin{aligned} \left. \frac{d^2}{d\omega^2} \left( B^{(2)}(\omega) \right) \right|_{2\pi\ell} &= \left. \frac{d}{d\omega} \left( \frac{3}{2} \left( \text{sinc}\left(\frac{\omega}{2}\right) \right)^2 \frac{d}{d\omega} \left( \text{sinc}\left(\frac{\omega}{2}\right) \right) \right) \right|_{2\pi\ell} \\ &= \frac{3}{2} \text{sinc}\left(\frac{\omega}{2}\right) \left( \frac{1}{2} \text{sinc}\left(\frac{\omega}{2}\right) \frac{d^2}{d\omega^2} \left( \text{sinc}\left(\frac{\omega}{2}\right) \right) + \right. \\ &\quad \left. + \left( \frac{d}{d\omega} \left( \text{sinc}\left(\frac{\omega}{2}\right) \right) \right)^2 \right) \Big|_{2\pi\ell} = 0, \end{aligned} \quad (5.72)$$

by the same arguments as before. Then, the theorem states that the linear combinations of  $\{\beta^{(2)}(t - n)\}_{n \in \mathbb{Z}}$  can reproduce polynomials up to degree 2, as we had already seen in Example 5.13 using a time-domain argument.

## 5.4 Approximation of Functions and Sequences by Series Truncation

In our discussion of sampling expansions in Chapter 4, sampling followed by interpolation creates an approximation in a predetermined subspace; that subspace  $S$  is determined by the interpolation operator, and the manner of projecting to  $S$  as orthogonal to a subspace  $\tilde{S}$  is determined by the sampling operator. The reconstruction is a series expansion using a basis for  $S$ . This setting was specifically geared at sampling/interpolation systems used in analog to digital conversion, and the subspace  $S$  is therefore typically a shift-invariant subspace.

We will now generalize this setting to consider approximation in any basis, orthogonal or biorthogonal, and even in frames. Given a class of functions or sequences, which are good bases to use in the approximation? How about the approximation method as well as the measure of approximation? We consider answers to these questions for linear and nonlinear approximations, in deterministic as well as stochastic settings. In all cases, approximation is by series truncation; the series in questions might be reordered initially and then truncated.

### 5.4.1 Linear Approximation

Consider a space  $S$ , for which we have an orthonormal basis  $\{\varphi_k\}_{k \in \mathbb{N}}$ . Given any  $x \in S$ , we can write it as the expansion (1.85a),

$$x = \sum_{k \in \mathbb{N}} \langle x, \varphi_k \rangle \varphi_k. \quad (5.73)$$

Consider now an  $M$ -term approximation,  $\hat{x}_M$ , as the orthogonal projection onto  $S_M$ , the space spanned by the first  $M$  basis vectors. The approximation, difference and the quadratic approximation error are

$$\hat{x}_M = \sum_{k=0}^{M-1} \langle x, \varphi_k \rangle \varphi_k, \quad (5.74a)$$

$$d_M = x - \hat{x}_M = \sum_{k=M}^{\infty} \langle x, \varphi_k \rangle \varphi_k, \quad (5.74b)$$

$$\epsilon_M^2 = \|d_M\|^2 = \|x - \hat{x}_M\|^2 = \sum_{k=M}^{\infty} |\langle x, \varphi_k \rangle|^2. \quad (5.74c)$$

From the projection theorem, Theorem 1.26, we know that the error of the approximation is orthogonal to the approximation itself,

$$d_M \perp \hat{x}_M, \quad (5.75a)$$

$$\|x\|^2 = \|\hat{x}_M\|^2 + \|d_M\|^2. \quad (5.75b)$$

Our statements so far where about orthonormal bases; with care, they can be extended to the biorthogonal ones (see Exercise 5.16).

The expressions above are all deterministic. A given  $x$  is approximated in a fixed basis with a deterministic algorithm, independent of  $x$ . Classes of objects are functions with certain characteristics, such as continuous functions. A stochastic version simply considers the approximation of the stochastic processes  $x$  as in (5.74a), leading to an expected quadratic error,

$$\mathbb{E}[\epsilon_M^2] = \mathbb{E}[\|x - \hat{x}_M\|^2]. \quad (5.76)$$

### 5.4.2 Nonlinear Approximation

Among a vast set of possible approximations beyond the linear case seen above, we consider a nonlinear approximation method in bases. Consider again a space  $S$ , for which we have an orthonormal basis  $\{\varphi_k\}_{k \in \mathbb{N}}$ , such that (5.73) holds for all  $x \in S$ . Now, suppose we can choose any  $M$  terms in the expansion (5.73), not necessarily the first  $M$  as in (5.74a). As we are expanding in an orthonormal basis, it is not hard to see that we should choose the  $M$  largest-magnitude coefficients so as to minimize the quadratic approximation error.

Start by creating an ordered sequence  $\{\alpha_{k_n}\}$  of the expansion coefficients

$$\alpha_k = \langle x, \varphi_k \rangle,$$

## 5.4. Approximation of Functions and Sequences by Series Truncation

495

such that  $|\alpha_{k_n}| \geq |\alpha_{k_{n+1}}|$  for  $n \in \mathbb{N}$  (when the inequality is not strict, the ordering is not unique). To choose the  $M$  largest-magnitude coefficients, define the set  $\mathcal{I}_M(x)$ ,

$$\mathcal{I}_M(x) = \{k_0, k_1, \dots, k_{M-1}\}, \quad (5.77)$$

where we made explicit that this set depends on  $x$ . Because of our choice of  $\mathcal{I}_M(x)$ ,

$$\sum_{n \in \mathcal{I}_M} |\alpha_n|^2 \geq \sum_{n \in \mathcal{J}_M} |\alpha_n|^2 \quad (5.78)$$

where  $\mathcal{J}_M$  is any other set of  $M$  indices. Call  $\hat{x}_M$  and  $\hat{x}'_M$  the projections of  $x$  onto the spaces spanned by  $\{\varphi_k\}_{k \in \mathcal{I}_M}$  and  $\{\varphi_k\}_{k \in \mathcal{J}_M}$ , respectively,

$$\hat{x}_M = \sum_{k \in \mathcal{I}_M} \langle x, \varphi_k \rangle \varphi_k, \quad (5.79a)$$

$$\hat{x}'_M = \sum_{k \in \mathcal{J}_M} \langle x, \varphi_k \rangle \varphi_k. \quad (5.79b)$$

Then  $\hat{x}_M$  is the best  $M$ -term approximation (possibly nonunique) of  $x$ . In other words,

$$\|x - \hat{x}_M\|^2 \leq \|x - \hat{x}'_M\|^2. \quad (5.80)$$

This is equivalent to

$$\sum_{k \notin \mathcal{I}_M} |\alpha_k|^2 \leq \sum_{k \notin \mathcal{J}_M} |\alpha_k|^2 \quad (5.81)$$

since we are in an orthonormal expansion. This is because

$$\begin{aligned} \|x\|^2 &= \sum_{n \in \mathcal{I}_M} |\alpha_n|^2 + \sum_{n \notin \mathcal{I}_M} |\alpha_n|^2 \\ &= \sum_{n \in \mathcal{J}_M} |\alpha_n|^2 + \sum_{n \notin \mathcal{J}_M} |\alpha_n|^2. \end{aligned}$$

From now on, we use the set  $\mathcal{I}_M(x)$  in (5.77) to obtain the best nonlinear approximation  $\hat{x}_M$  in (5.79a). The difference and the quadratic approximation error are

$$d_M = x - \hat{x}_M = \sum_{k \notin \mathcal{I}_M} \langle x, \varphi_k \rangle \varphi_k, \quad (5.82a)$$

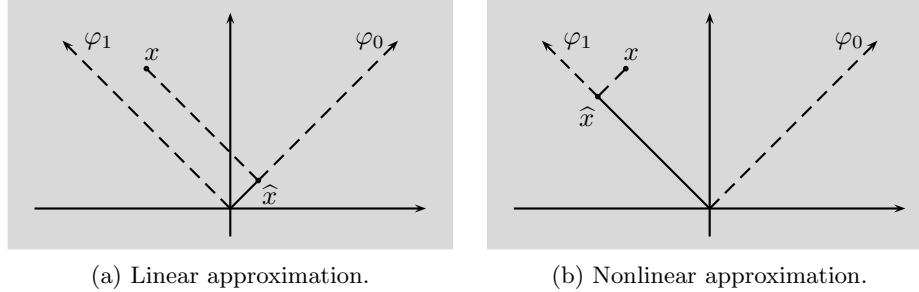
$$\epsilon_M^2 = \|d_M\|^2 = \|x - \hat{x}_M\|^2 = \sum_{k \notin \mathcal{I}_M} |\langle x, \varphi_k \rangle|^2. \quad (5.82b)$$

As before, the error of the approximation is orthogonal to the approximation itself,

$$d_M \perp \hat{x}_M, \quad (5.83a)$$

$$\|x\|^2 = \|\hat{x}_M\|^2 + \|d_M\|^2. \quad (5.83b)$$

This approximation method is nonlinear because if  $\hat{x}_M$  and  $\hat{y}_M$  are the approximations of  $x$  and  $y$  according to (5.79a), then the approximation of  $x + y$  is in general



**Figure 5.25:** One-dimensional approximation in  $\mathbb{R}^2$ . The orthonormal basis is  $\varphi_0 = [1\ 1]^T/\sqrt{2}$  and  $\varphi_1 = [-1\ 1]^T/\sqrt{2}$ . (a) Linear approximation by keeping only the first coefficient, or  $S = \alpha\varphi_0$ . (b) Nonlinear approximation by keeping the largest-magnitude coefficient. The first and third quadrants will be approximated by  $\varphi_0$ , the rest by  $\varphi_1$ .

not equal to  $\hat{x}_M + \hat{y}_M$ . This is easy to see, because the sets of largest-magnitude coefficients are usually different,

$$\mathcal{I}_M(x) \neq \mathcal{I}_M(y),$$

that is, the approximation depends on the signal we wish to approximate. The difference between linear and nonlinear approximations is illustrated in Figure 5.25; see also Exercise 5.17.

Interestingly, the difference between linear and nonlinear approximations can be substantial, depending on the class of signals and the type of basis. As we know, Fourier bases are not good at approximating signals containing discontinuities, be it using linear or nonlinear approximation, due to the Gibbs phenomenon. Changing the representation basis to wavelets, and using nonlinear instead of linear approximation totally changes the game, as illustrated in Figure 5.5 (linear approximation using Fourier series) and Figure 5.6 (nonlinear approximation using Haar basis).

### 5.4.3 Approximation in Fourier Bases

When approximating a function  $x(t)$  either by linear or nonlinear approximation, the key is the decay of the expansion coefficients, either in their natural ordering in the linear case, or reordered given by  $\mathcal{I}_M(x)$  in (5.77) in the nonlinear case.

As we have seen for both the Fourier transform in (3.79) and Fourier series in (3.116), smoothness of the time-domain function and the decay of the Fourier transform or Fourier series coefficients are closely connected. We focus here on Fourier series, since it provides an orthonormal basis for one period of periodic functions. The condition in (3.116) is equivalent to

$$\sum_{k \in \mathbb{Z}} |k|^p |X_k| < \infty \quad \Leftrightarrow \quad |X_k| \leq \frac{\gamma}{1 + |k|^{p+1+\epsilon}} \quad (5.84)$$

or,  $x(t)$  is  $p$ -times continuously differentiable. The converse, similarly to (3.79c), is



## 5.4. Approximation of Functions and Sequences by Series Truncation

497

that if  $x(t)$  is  $p$ -times continuously differentiable, then  $X_k$  decays at least as

$$|X_k| < \frac{\gamma}{1 + |k|^{p+1}}. \quad (5.85)$$

These decay rates allow us to bound the approximation error in the Fourier series, as we now illustrate.

**EXAMPLE 5.15 (LINEAR AND NONLINEAR APPROXIMATION IN FOURIER SERIES)**

Let  $x(t)$  be the box function of period 1, width  $1/2$ , norm 1, and centered at the origin. According to Table 3.6,  $x(t)$  and its Fourier series are given by

$$\begin{cases} \sqrt{2}, & |t| \leq 1/4; \\ 0, & \text{otherwise,} \end{cases} \quad \xleftrightarrow{\text{FS}} \quad \frac{1}{\sqrt{2}} \operatorname{sinc}(\pi k/2).$$

This Fourier series is symmetric around the origin, and all even terms (except at the origin) are zero,

$$X = \sqrt{2} \left[ \dots \quad 0 \quad -\frac{1}{3\pi} \quad 0 \quad \frac{1}{\pi} \quad \boxed{\frac{1}{2}} \quad \frac{1}{\pi} \quad 0 \quad -\frac{1}{3\pi} \quad 0 \quad \dots \right].$$

Using linear approximation, we keep the central  $M = 4K - 1$  terms, with the quadratic error as

$$\epsilon_M^2 = \frac{4}{\pi^2} \sum_{|k| \geq K} \frac{1}{(2k+1)^2}.$$

By approximating this sum with an integral, we see that the quadratic error decays as  $\gamma/M$ . Choosing nonlinear approximation instead, we can skip all the zero terms; this will improve the approximation constant, but not the order, as the quadratic error still decays as  $\gamma_1/M$ .

In general, the quadratic error for an  $M$ -term approximation of the Fourier series of  $p$ -times differentiable  $x(t)$ , whose Fourier series decays as in (5.85), is of the order

$$\sum_{n=M}^{\infty} \frac{\gamma_2}{1 + |k|^{2p+2}} \sim \frac{\gamma_3}{M^{2p+1}}.$$

Moreover, this holds both for linear and nonlinear approximation, as we have just seen in Example 5.15.

**5.4.4 Karhunen-Loève Transform**

A cornerstone result for the linear approximation of stochastic processes is the optimality of the Karhunen-Loève transform. The result can be derived in many forms, but they all hinge on calculating the eigendecomposition of the autocorrelation of the process, and keeping the largest eigenvalues in that decomposition.

**KLT for Random Vectors** Let  $\mathbf{x} = [x_0 \ x_1 \ \dots \ x_{N-1}]^T$  be an  $N$ -dimensional random vector. For convenience, we assume all  $x_k$  to have mean zero, and we denote the autocorrelation matrix of  $\mathbf{x}$  by

$$A = E[\mathbf{x}\mathbf{x}^*]. \quad (5.86)$$

We want to answer the following question: What is the best orthonormal basis for  $\mathbb{C}^N$ , for which a linear approximation on a subspace of any  $M < N$  minimizes the expected quadratic error? In other words, search for a set of orthonormal vectors  $\{\varphi_k\}_{k=0}^{N-1}$  forming a basis for  $\mathbb{C}^N$ , such that the quadratic error between  $\mathbf{x}$  and its approximation  $\hat{\mathbf{x}}$ , given below, is minimized,

$$\mathbf{x} = \sum_{k=0}^{N-1} \langle \mathbf{x}, \varphi_k \rangle \varphi_k, \quad \hat{\mathbf{x}} = \sum_{k=0}^{M-1} \langle \mathbf{x}, \varphi_k \rangle \varphi_k. \quad (5.87)$$

This quadratic error is

$$\begin{aligned} \epsilon_M^2 &= E[\|\mathbf{x} - \hat{\mathbf{x}}\|^2] = E[\langle \mathbf{x} - \hat{\mathbf{x}}, \mathbf{x} - \hat{\mathbf{x}} \rangle] \\ &\stackrel{(a)}{=} E\left[\left\langle \sum_{k=M}^{N-1} \langle \mathbf{x}, \varphi_k \rangle \varphi_k, \sum_{m=M}^{N-1} \langle \mathbf{x}, \varphi_m \rangle \varphi_m \right\rangle\right] \\ &\stackrel{(b)}{=} E\left[\sum_{k=M}^{N-1} \sum_{m=M}^{N-1} \langle \langle \mathbf{x}, \varphi_k \rangle \varphi_k, \langle \mathbf{x}, \varphi_m \rangle \varphi_m \rangle\right] \\ &\stackrel{(c)}{=} E\left[\sum_{k=M}^{N-1} \sum_{m=M}^{N-1} \langle \mathbf{x}, \varphi_k \rangle \langle \mathbf{x}, \varphi_m \rangle^* \langle \varphi_k, \varphi_m \rangle\right] \\ &\stackrel{(d)}{=} E\left[\sum_{k=M}^{N-1} \sum_{m=M}^{N-1} \langle \mathbf{x}, \varphi_k \rangle \langle \varphi_m, \mathbf{x} \rangle \delta_{k-m}\right] \\ &\stackrel{(e)}{=} E\left[\sum_{k=M}^{N-1} \langle \mathbf{x}, \varphi_k \rangle \langle \varphi_k, \mathbf{x} \rangle\right] \stackrel{(f)}{=} E\left[\sum_{k=M}^{N-1} \varphi_k^* \mathbf{x} \mathbf{x}^* \varphi_k\right] \stackrel{(g)}{=} \varphi_k^* E[\mathbf{x} \mathbf{x}^*] \varphi_k \\ &\stackrel{(h)}{=} \sum_{k=M}^{N-1} \varphi_k^* A \varphi_k \stackrel{(i)}{=} \sum_{k=M}^{N-1} \langle A \varphi_k, \varphi_k \rangle, \end{aligned} \quad (5.88)$$

where (a) follows from (5.87); (b) from the linearity of the inner product; (c) from linearity of the inner product in the first argument and conjugate linearity in the second argument; (d) from the orthonormality of the basis; (e) from the sum in  $m$  being nonzero only for  $m = k$ ; in (f) we wrote the inner products in vector notation; (g) from the linearity of expectation; (h) from the definition of the autocorrelation matrix (5.86); and in (g) we used inner-product notation again.

**THEOREM 5.8 (KARHUNEN–LOÈVE TRANSFORM)** For  $1 \leq M < N$ , the expected error between  $\mathbf{x}$  and its linear approximation on a subspace of dimension  $M$

is minimized by the basis  $\{\varphi_k\}_{k=0}^{N-1}$  consisting of eigenvectors of  $A$  ordered by decreasing eigenvalues,

$$A\varphi_k = \lambda_k\varphi_k, \quad k = 0, 1, \dots, N-1, \quad (5.89a)$$

$$\lambda_k \geq \lambda_{k+1}, \quad k = 0, 1, \dots, N-2. \quad (5.89b)$$

*Proof.* We briefly sketch the proof. Minimizing (5.88) is equivalent to maximizing

$$\sum_{k=0}^{M-1} \langle A\varphi_k, \varphi_k \rangle \quad (5.90)$$

because the basis is orthonormal. Since  $A$  is positive semidefinite, all its eigenvalues are real and nonnegative. Assume for simplicity that the eigenvalues are distinct. Starting with  $M = 1$ , choose  $\varphi_0$  such that

$$\varphi_0 = \arg \max_{\|\varphi_0\|=1} \langle A\varphi_0, \varphi_0 \rangle,$$

that is, the eigenvector corresponding to the largest eigenvalue. Assuming the first  $K$  vectors have been chosen so as to maximize  $\sum_{k=0}^{K-1} \langle A\varphi_k, \varphi_k \rangle$ , how do we choose  $\varphi_K$ ? It has to be of norm 1 and orthogonal to the span( $\{\varphi_0, \varphi_1, \dots, \varphi_{K-1}\}$ ). This maximization leads to the next eigenvector with the largest eigenvalue. If eigenvalues are not distinct, then one can take any norm-1 linear combination of the eigenvectors corresponding to multiplicities.

**KLT for WSS Processes** We now extend the idea of the KLT from random vectors to WSS processes. Assume  $x_n$  is WSS with zero mean and autocorrelation  $a_n$  with sufficient decay so that  $a \in \ell^1(\mathbb{Z})$ . The power spectral density is (see (2.232))

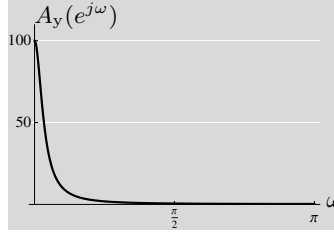
$$a_n \xleftrightarrow{\text{DTFT}} A(e^{j\omega}), \quad (5.91)$$

which is positive semidefinite. Here, the autocorrelation matrix  $A$  is an infinite-dimensional Toeplitz matrix with diagonals given by  $a_n$ ,

$$A = E[xx^*] = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \ddots \\ \dots & a_0 & a_1 & a_2 & \dots \\ \dots & a_1 & \boxed{a_0} & a_1 & \dots \\ \dots & a_2 & a_1 & a_0 & \dots \\ \ddots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \quad (5.92)$$

This Toeplitz operator has a continuous power spectral density given by  $A(e^{j\omega})$  (rather than a discrete set of eigenvalues), and under the  $\ell^1(\mathbb{Z})$  assumption, the eigensequences are the DTFT sequences  $e^{j\omega k}$ ,  $k \in \mathbb{Z}$ ,  $\omega \in [0, 2\pi)$ . The subspace approximation of size  $M$  in  $\mathbb{R}^N$  becomes a subset approximation of the spectrum, namely a set  $S \subset [0, 2\pi)$  such that

$$\begin{aligned} S & \quad \text{such that} \quad \frac{|S|}{2\pi} = \frac{M}{N}, \\ \omega_0 \in S & \quad \text{if} \quad A(e^{j\omega_0}) \geq A(e^{j\omega}), \text{ for } \omega \notin S. \end{aligned}$$



**Figure 5.26:** KLT for an AR-1 process is the truncation of the monotonically decreasing power spectral density  $A_y(e^{j\omega})$  from (5.93) to  $[-\pi/2, \pi/2]$ .

We illustrate this on an AR-1 process.

**EXAMPLE 5.16 (KLT FOR AN AR-1 PROCESS)** Take an i.i.d. process  $x_n$  as input to an AR-1 model with transfer function  $1/(1 - \alpha z^{-1})$ . According to (2.234), the power spectral density of the filtered process  $y_n$  is

$$A_y(e^{j\omega}) = \frac{1}{(1 - \alpha e^{-j\omega})(1 - \alpha e^{j\omega})} = \frac{1}{1 + \alpha^2 - 2\alpha \cos \omega}$$

(see Example 4.25). Choose  $\alpha = 0.9$ . The power spectral density is

$$A_y(e^{j\omega}) = \frac{1}{1.81 - 1.8 \cos \omega} \quad (5.93)$$

and varies from 100 at  $\omega = 0$  to  $(3.61)^{-1} \approx 0.277$  at  $\omega = \pi$ . Since the power spectral density is monotonically decreasing, it is easy to find a set  $S$  of any given size. Choose  $|S| = \pi$ . The KLT is the projection of  $y_n$  onto  $\text{BL}[-\pi/2, \pi/2]$ , which we have seen in Example 4.25, see Figure 5.26.

## 5.5 Compression and Transform Coding

The previous sections concentrated on approximating a given function or a sequence by using a subset of expansion coefficients. In this section, we address issue (iii) from the beginning of this chapter, that is, what to do when the chosen expansion coefficients require too much storage or bandwidth for transmission. In other words, we compress.

Everyday compression problems are unmanageable without a divide and conquer approach.<sup>94</sup> Effective compression of images, for example, depends on the tendencies of pixels to be similar to their neighbors, or to differ in partially predictable ways. These tendencies, arising from the continuity, texturing, and boundaries of objects, the similarity of objects in an image, gradual lighting changes, an artist's

<sup>94</sup>Divide and conquer approach is central to many endeavors, ranging from children managing inconsistent parents to colonial powers controlling native peoples. In engineering and computational science, it means breaking a big problem into smaller problems that can be more easily understood and solved. Putting the pieces back together gives a modular design, which is advantageous for implementation, testing, and component reuse.

technique and color palette, etc., may extend over an entire image with a quarter million pixels. Yet the most general way to utilize the probable relationships between pixels (later described as *unconstrained source coding*) is infeasible for this many pixels. In fact, 16 pixels is a lot for an unconstrained source code.

To conquer the compression problem—allowing, for example, more than 16 pixels to be encoded simultaneously—state-of-the-art lossy compressors divide the encoding operation into a sequence of three relatively simple steps: the computation of a linear transformation of the data designed primarily to produce uncorrelated coefficients, separate quantization of each scalar coefficient, and entropy coding. This process is called *transform coding*. In image compression, a square image with  $N$  pixels is typically processed with simple linear transforms (often the DCTs or DWTs) of size  $\sqrt{N}$ .

This section explains the fundamental principles of transform coding; these principles apply equally well to images, audio, video, and various other types of data, so abstract formulations are given. For more details, see *Further Reading*.

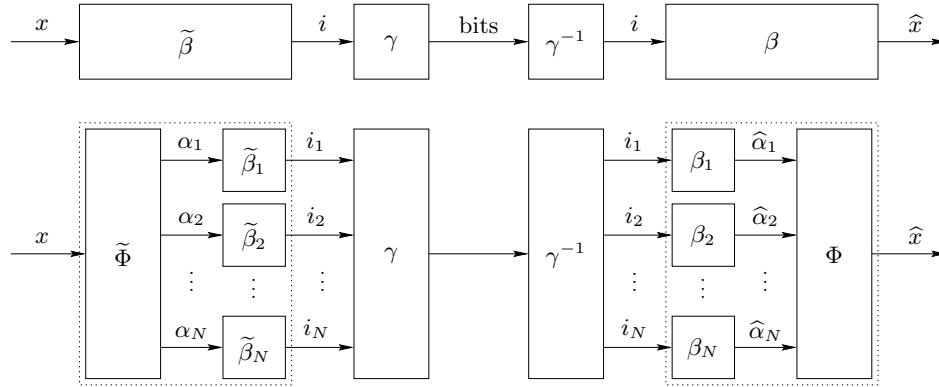
**Source Coding** *Source coding* means to represent information in bits, with the natural aim of using a small number of bits. When the information can be exactly recovered from the bits, the source coding or *compression* is called *lossless*; otherwise, it is called *lossy*. The transform codes in this section are lossy. However, lossless entropy codes appear as components of transform codes, so both lossless and lossy compression are of present interest.

In our discussion, the “information” is denoted by a real column vector  $x \in \mathbb{R}^N$  or a sequence of such vectors. A vector might be formed from pixel values in an image or by sampling an audio signal;  $KN$  pixels can be arranged as a sequence of  $K$  vectors of length  $N$ . The vector length  $N$  is defined such that each vector in a sequence is encoded independently. For the purpose of building a mathematical theory, the source vectors are assumed to be realizations of a random vector  $x$  with a known distribution. The distribution could be purely empirical.

A source code is comprised of two mappings: an *encoder* and a *decoder*. The encoder maps any vector  $x \in \mathbb{R}^N$  to a finite string of bits, and the decoder maps any of these strings of bits to an approximation  $\hat{x} \in \mathbb{R}^N$ . The encoder mapping can always be factored as  $\gamma \circ \tilde{\beta}$ , where  $\tilde{\beta}$  is a mapping from  $\mathbb{R}^N$  to some discrete set  $\mathcal{I}$  and  $\gamma$  is an invertible mapping from  $\mathcal{I}$  to strings of bits. The former is called a lossy encoder and the latter a lossless code or an entropy code. The decoder inverts  $\gamma$  and then approximates  $x$  from the index  $\tilde{\beta}(x) \in \mathcal{I}$ . This is shown in the top half of Figure 5.27. It is assumed that communication between the encoder and decoder is perfect.

To assess the quality of a lossy source code, we need numerical measures of approximation accuracy and description length. The measure for description length is simply the expected number of bits output by the encoder divided by  $N$ ; this is called the *rate* in bits per scalar sample and denoted by  $R$ . Here we will measure approximation accuracy by squared Euclidean norm divided by the vector length:

$$d(x, \hat{x}) = \frac{1}{N} \|x - \hat{x}\|^2 = \frac{1}{N} \sum_{n=0}^{N-1} (x_n - \hat{x}_n)^2.$$



**Figure 5.27:** Any source code can be decomposed so that the encoder is  $\gamma \circ \tilde{\beta}$  and the decoder is  $\beta \circ \gamma^{-1}$ , as shown at top.  $\gamma$  is an entropy code and  $\tilde{\beta}$  and  $\beta$  are the encoder and decoder of an  $N$ -dimensional quantizer. In a transform code,  $\tilde{\beta}$  and  $\beta$  each have a particular constrained structure. In the encoder,  $\tilde{\beta}$  is replaced with a linear transform  $\tilde{\Phi}$  and a set of  $N$  scalar quantizer encoders. The intermediate  $\alpha_i$ 's are the expansions coefficients, here called here transform coefficients. In the decoder,  $\beta$  is replaced with  $N$  scalar quantizer decoders and another linear transform  $\Phi$ . Usually  $\Phi = \tilde{\Phi}^{-1}$ . [TfBD: Split into parts (a) and (b).]

This accuracy measure is conventional and usually leads to the easiest mathematical results, though the theory of source coding has been developed with quite general measures [10]. The expected value of  $d(x, \hat{x})$  is the MSE *distortion* from (1.65) and is denoted by  $D = E[d(x, \hat{x})]$ . The normalizations by  $N$  make it possible to fairly compare source codes with different lengths.

Fixing  $N$ , a theoretical concept of optimality is straightforward: A length- $N$  source code is *optimal* if no other length- $N$  source code with at most the same rate has lower distortion. This concept is of dubious value. First, it is very difficult to check the optimality of a source code. Local optimality—being assured that small perturbations of  $\tilde{\beta}$  and  $\beta$  will not improve performance—is often the best that can be attained. Second, and of more practical consequence, a system designer gets to choose the value of  $N$ . It can be as large as the total size of the data set—like the number of pixels in an image—but can also be smaller, in which case the data set is interpreted as a sequence of vectors.

There are conflicting motives in choosing  $N$ . Compression performance is related to the predictability of one part of  $x$  from the rest. Since predictability can only increase from having more data, performance is usually improved by increasing  $N$ . The conflict comes from the fact that the computational complexity of encoding is also increased. This is particularly dramatic if one looks at complexities of optimal source codes. The obvious way to implement an optimal encoder is to search through the entire codebook, giving running time exponential in  $N$ .

State-of-the-art source codes result from an intelligent compromise; instead of attempting to realize an optimal code for a given value of  $N$  whose encoding complexity would force a small value for  $N$ , source codes that are good, but not

optimal, are used. Their lower complexities make much larger  $N$ 's feasible. The paradoxical conclusion is that the best codes to use in practice are *suboptimal*.

**Constrained Source Coding** Transform codes are the most used source codes because they are easy to apply at any rate and even with very large values of  $N$ . The essence of transform coding is the modularization shown in the bottom half of Figure 5.27. The mapping  $\tilde{\beta}$  is implemented in two steps. First, an invertible linear transform of the source vector  $x$  is computed, producing  $\alpha = \Phi x$ . These are the expansion coefficients we have seen earlier; in coding theory, they are called *transform coefficients*. The  $N$  expansion coefficients are then quantized independently of each other by  $N$  scalar quantizers. This is called *scalar quantization* since each scalar component of  $\alpha$  is treated separately. Finally, the quantizer indices that correspond to the transform coefficients are compressed with an entropy code to produce the sequence of bits that represent the data.

To reconstruct an approximation of  $x$ , the decoder essentially reverses the steps of the encoder. The action of the entropy coder can be inverted to recover the quantizer indices. Then the decoders of the scalar quantizers produce a vector  $\hat{y}$  of estimates of the expansion coefficients. To complete the reconstruction, a linear transform is applied to  $\hat{y}$  to produce the approximation  $\hat{x}$ . This final step usually uses the transform  $\tilde{\Phi}^{-1}$ , but for generality the transform is denoted  $\Phi$ .

Most source codes cannot be implemented in the two stages of linear transform and scalar quantization. Thus, a transform code is an example of a *constrained source code*. Constrained source codes are, loosely speaking, source codes that are suboptimal but have low complexity. The simplicity of transform coding allows large values of  $N$  to be practical. Computing the transform  $\Phi$  requires at most  $N^2$  multiplications and  $N(N-1)$  additions. Specially structured transforms, such as the DFT, DCT, or DWT, are often used to reduce the complexity of this step, but this is merely icing on the cake. The great reduction from the exponential complexity of a general source code to the (at most) quadratic complexity of a transform code comes from using linear transforms and scalar quantization.

### 5.5.1 Transform Coding

The standard theoretical model for transform coding looks like the bottom of Figure 5.27. It has the strict modularity shown, meaning that the transform, quantization and entropy coding blocks operate independently. In addition, the entropy coder can be decomposed into  $N$  parallel entropy coders so that the quantization and entropy coding operate independently on each scalar transform coefficient.

We start by briefly describing the fundamentals of entropy coding and quantization to provide background for our later focus on the optimization of the transform, and then address the allocation of bits among the  $N$  scalar quantizers.

#### Entropy Coding

Entropy codes are used for lossless coding of discrete random variables. Consider the discrete random variable  $x$  with alphabet  $\mathcal{I}$ . An entropy code  $\gamma$  assigns a unique

binary string, called a *codeword*, to each  $i \in \mathcal{I}$  (see Figure 5.27).

Since the codewords are unique, an entropy code is always invertible. However, we will place more restrictive conditions on entropy codes so they can be used on sequences of realizations of  $\mathbf{x}$ . The *extension* of  $\gamma$  maps the finite sequence  $[x_1 \ x_2 \ \dots \ x_k]$  to the concatenation of the outputs of  $\gamma$  with each input,  $\gamma(x_1)\gamma(x_2)\dots\gamma(x_k)$ . A code is called *uniquely decodable* if its extension is one-to-one. A uniquely decodable code can be applied to message sequences without adding any “punctuation” to show where one codeword ends and the next begins. For example, in a *prefix code*, no codeword is the prefix of any other codeword. Prefix codes are guaranteed to be uniquely decodable.

A trivial code numbers each element of  $\mathcal{I}$  with a distinct index in  $\{0, 1, \dots, |\mathcal{I}|-1\}$  and maps each element to the binary expansion of its index. Such a code requires  $\lceil \log_2 |\mathcal{I}| \rceil$  bits per symbol. This is considered the *lack* of an entropy code. The idea in entropy code design is to minimize the mean number of bits used to represent  $\mathbf{x}$  at the expense of making the worst-case performance worse. The expected code length is given by

$$L(\gamma) = \mathbb{E}[\ell(\gamma(\mathbf{x}))] = \sum_{i \in \mathcal{I}} P_{\mathbf{x}}(i) \ell(\gamma(i)),$$

where  $P_{\mathbf{x}}(i)$  is the probability of symbol  $i$  and  $\ell(\gamma(i))$  is the length of  $\gamma(i)$ . The expected length can be reduced if short code words are used for the most probable symbols, even if that means that some symbols will have codewords with more than  $\lceil \log_2 |\mathcal{I}| \rceil$  bits.

The entropy code  $\gamma$  is called *optimal* if it is a prefix code that minimizes  $L(\gamma)$ . Huffman codes, described shortly, are examples of optimal codes. The performance of an optimal code is bounded by

$$H(\mathbf{x}) \leq L(\gamma) < H(\mathbf{x}) + 1 \quad (5.94)$$

where

$$H(\mathbf{x}) = - \sum_{i \in \mathcal{I}} P_{\mathbf{x}}(i) \log_2 P_{\mathbf{x}}(i) \quad (5.95)$$

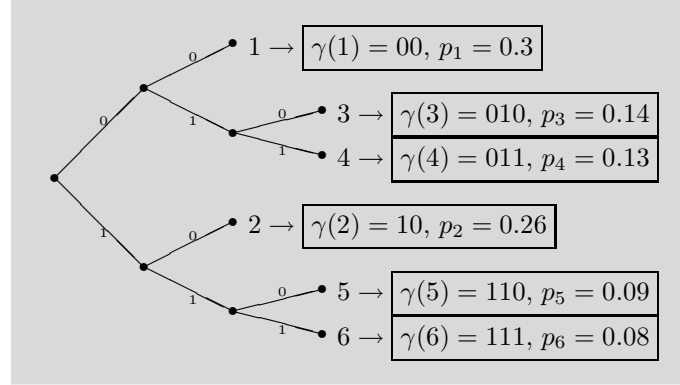
is the *entropy* of  $\mathbf{x}$ .

The up to one bit gap in (5.94) is ignored in the remainder of the section. If  $H(\mathbf{x})$  is large, this is justified simply because one bit is small compared to the code length. Otherwise note that  $L(\gamma) \approx H(\mathbf{x})$  can be attained by coding blocks of symbols together; this is detailed in any information theory or data compression textbook.

**Huffman Coding** There is a simple algorithm, due to Huffman [78], for constructing optimal entropy codes. One starts with a graph with one node for each symbol and no edges. These nodes will become the leaves of a tree as edges are added to make a connected graph.

At each step of the algorithm, the probabilities of the disconnected sets of nodes are sorted and the two least probable sets are merged through the addition of a parent node and edges to each of the two sets. The edges are assigned labels



**Figure 5.28:** Huffman code.

of 0 and 1. When a tree has been formed, codewords are assigned to each leaf node by concatenating the edge labels on the path from the root to the leaf.

**EXAMPLE 5.17 (HUFFMAN CODING)** Figure 5.28 shows a Huffman code tree for symbols  $\{1, 2, 3, 4, 5, 6\}$  with respective probabilities  $\{0.3, 0.26, 0.14, 0.13, 0.09, 0.08\}$ . The codewords are boxed. Computing a weighted sum of the codeword lengths gives the expected code length

$$L = 0.3 \cdot 2 + 0.26 \cdot 2 + 0.14 \cdot 3 + 0.13 \cdot 3 + 0.09 \cdot 3 + 0.08 \cdot 3 = 2.44 \text{ bits.}$$

This is quite close to the entropy of 2.41 bits obtained by evaluating (5.95).

### Quantization

A quantizer  $q$  is a mapping from a source alphabet  $\mathbb{R}^N$  to a *reproduction codebook*  $\mathcal{C} = \{\hat{x}_i\}_{i \in \mathcal{I}} \subset \mathbb{R}^N$ , where  $\mathcal{I}$  is an arbitrary countable index set. Quantization can be decomposed into two operations  $q = \beta \circ \tilde{\beta}$ , as shown in Figure 5.27. The *lossy encoder*  $\tilde{\beta} : \mathbb{R}^N \rightarrow \mathcal{I}$  is specified by a partition of  $\mathbb{R}^N$  into *partition cells*  $S_i = \{x \in \mathbb{R}^N \mid \tilde{\beta}(x) = i\}$ ,  $i \in \mathcal{I}$ . The *reproduction decoder*  $\beta : \mathcal{I} \rightarrow \mathbb{R}^N$  is specified by the codebook  $\mathcal{C}$ . If  $N = 1$ , the quantizer is called a *scalar quantizer*; for  $N > 1$ , it is a *vector quantizer*.

The quality of a quantizer is determined by its distortion and rate. The MSE distortion for quantizing random vector  $\mathbf{x} \in \mathbb{R}^N$  is

$$D = \frac{1}{N} \mathbb{E}[\|\mathbf{x} - q(\mathbf{x})\|^2].$$

The rate can be measured in a few ways. The lossy encoder output  $\tilde{\beta}(\mathbf{x})$  is a discrete random variable that is typically entropy coded because the output symbols will have unequal probabilities. Associating an entropy code  $\gamma$  to the quantizer gives a *variable-rate quantizer* specified by  $(\tilde{\beta}, \beta, \gamma)$ . The rate of the quantizer is

Quantizer	Rate $R$
Variable rate	$N^{-1}L(\gamma)$
Fixed rate	$N^{-1}\log_2  \mathcal{I} $
Entropy constrained	$N^{-1}H(\tilde{\beta}(x))$

**Table 5.2:** Rates for different quantizers.

the expected code length of  $\gamma$  divided by  $N$ . Not specifying an entropy code (or specifying the use of fixed-rate binary expansion) gives a *fixed-rate quantizer* with rate  $R = N^{-1}\log_2 |\mathcal{I}|$ . Measuring the rate by the idealized performance of an entropy code gives  $R = N^{-1}H(\tilde{\beta}(x))$ ; the quantizer in this case is called *entropy-constrained*. These rates are summarized in Table 5.2.

While the optimal performance of entropy-constrained quantization is better than that of variable-rate quantization as well as fixed-rate quantization, it adds complexity, and variable-length output can create difficulties such as buffer overflows. Furthermore, entropy-constrained quantization is only an idealization since the code will in general not meet the lower bound in (5.94).

**Optimal Quantization** An optimal quantizer is one that either minimizes the distortion subject to an upper bound on the rate, or minimizes the rate subject to an upper bound on the distortion. Because of simple shifting and scaling properties, an optimal quantizer for a random variable  $x$  can be easily deduced from an optimal quantizer for the normalized random variable  $(x - \mu_x)/\sigma_x$ , where  $\mu_x$  and  $\sigma_x$  are the mean and standard deviation of  $x$ , respectively. One consequence of this is that optimal quantizers have performance

$$D = \sigma_x^2 g(R), \quad (5.96)$$

where  $g(R)$  is the performance of optimal quantizers for the normalized source. Equation (5.96) holds, with a different function  $g$ , for any family of quantizers that can be described by its operation on a normalized variable, not just optimal quantizers.

The rate measure affects the optimal encoding rule because  $\tilde{\beta}(x)$  should be the index that minimizes a Lagrangian cost function including both rate and distortion. Only for fixed-rate quantization does the optimal encoding rule simplify to finding the index corresponding to the nearest codeword.

In some of the more technical discussions that follow, one property of optimal decoding is relevant: The optimal decoder  $\beta$  computes

$$\beta(i) = E[x \mid x \in S_i],$$

which is called *centroid reconstruction*. The conditional mean of the cell, or centroid, is the minimum MSE estimate (see Section 1.4.4).

**High-Resolution Quantization** For most sources, it is impossible to analytically express the performance of optimal quantizers. Thus, aside from using (5.96), ap-

proximations must suffice. Fortunately, approximations obtained when it is assumed that the quantization is very fine are reasonably accurate even at low to moderate rates.

High-resolution analysis is based on approximating the PDF  $f_x$  on the interval  $S_i$  by its value at the midpoint. Assuming  $f_x$  is smooth, this approximation is accurate when each  $S_i$  is short. Optimization of scalar quantizers then turns into finding the optimal lengths for the  $S_i$ 's, depending on the PDF  $f_x$ .

The performance of optimal fixed-rate quantization is approximately

$$D \approx \frac{1}{12} \left( \int_{\mathbb{R}} f_x^{1/3}(t) dt \right)^3 2^{-2R}. \quad (5.97)$$

For a Gaussian random variable whose PDF was given in (1.236) yields

$$D \approx \frac{\sqrt{3}\pi}{2} \sigma^2 2^{-2R}. \quad (5.98)$$

For entropy-constrained quantization, high-resolution analysis shows that it is optimal for each  $S_i$  to have equal length. A quantizer that partitions with equal-length intervals is called *uniform*.<sup>95</sup> The resulting performance is

$$D \approx \frac{1}{12} 2^{2h(x)} 2^{-2R}, \quad (5.99)$$

where

$$h(x) = - \int_{\mathbb{R}} f_x(t) \log_2 f_x(t) dt$$

is the *differential entropy* of  $x$ . For a Gaussian random variable, (5.99) simplifies to

$$D \approx \frac{\pi e}{6} \sigma^2 2^{-2R}. \quad (5.100)$$

Summarizing (5.97)–(5.100), we see that high-resolution quantizer performance is described by

$$D \approx c \sigma^2 2^{-2R}, \quad (5.101)$$

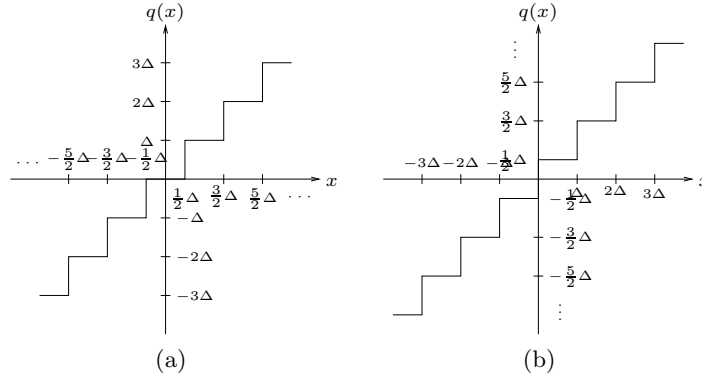
where  $\sigma^2$  is the variance of the source and  $c$  is a constant that depends on the normalized density of the source and the type of quantization (fixed-rate, variable-rate or entropy-constrained). This is consistent with (5.96).

The computations we have made are for scalar quantization. For vector quantization, the best performance in the limit as the dimension  $N$  grows is given by the distortion rate function. For a Gaussian source, this bound is

$$D = \sigma^2 2^{-2R}. \quad (5.102)$$

The approximate performance given by (5.100) is only worse by a factor of  $\pi e/6$  ( $\approx 1.53$  dB). This can be expressed as a redundancy  $\frac{1}{2} \log_2(\frac{\pi e}{6}) \approx 0.255$  bits. Furthermore, a numerical study has shown that for a wide range of memoryless sources, the redundancy of entropy-constrained uniform quantization is at most 0.3 bits per sample at all rates.

<sup>95</sup>While the design of quantizers has a deep theory, the fact remains that: “Most quantizers today are indeed uniform and scalar, but are combined with prediction or transforms.” [64].



**Figure 5.29:** Uniform quantization. (a) Rounding to the nearest integer. (b) Rounding to the nearest half-integer.

**Uniform Quantization** Though definitions of uniform quantization vary somewhat, the archetype of rounding is always an example of uniform quantization. Shown in Figure 5.29(a) is the input-output relationship of a device that rounds to the nearest integer multiple of the *step size*  $\Delta$ . To describe this in formal notation, the encoder could be  $\tilde{\beta}(x) = \text{round}(x/\Delta)$ , where  $\text{round}(\cdot)$  denotes rounding to the nearest integer, with the corresponding decoder  $\beta(i) = i\Delta$ .

The other common uniform quantizer is shown in Figure 5.29(b). This is a shifted version of the previous uniform quantizer. Variations in the definition of uniform quantization sometimes allow only the encoder to have equal length cells or only the decoder to have evenly spaced outputs and may also allow the decoder outputs to be shifted from the centers of the partition cells.

Uniform quantization of a uniform random variable provides a setting to see the typical trade-off between rate and distortion. Consider  $x$  uniformly distributed on the interval  $[0, 1)$  with the PDF as in (1.235a). A fixed-rate uniform quantizer, as in Figure 5.29(b), with  $K$  cells and step size  $\Delta = 1/K$ , quantizes  $x \in [(m-1)\Delta, m\Delta)$  to  $(m-\frac{1}{2})\Delta$  for  $m = 1, 2, \dots, K$ . It has rate  $R = \log_2 N = -\log_2 \Delta$  and distortion. Its rate and distortion are

$$R = \log_2 N = -\log_2 \Delta, \quad (5.103a)$$

$$\begin{aligned} D &= \int_0^1 (x - \hat{x})^2 dx = \sum_{n=0}^{N-1} \int_{(n-1)\Delta}^{n\Delta} (x - (n - \frac{1}{2})\Delta)^2 dx \\ &= \frac{1}{12}\Delta^2 = \frac{1}{12}2^{-2R}. \end{aligned} \quad (5.103b)$$

We see that when using the MSE distortion measure, the  $2^{-2R}$  factor will almost always be present.

### Bit Allocation

Coding (quantizing and entropy coding) each expansion coefficient separately splits the total number of bits among the transform coefficients in some manner, implying

## 5.5. Compression and Transform Coding

509

some sort of a *bit allocation* among the components.

Bit allocation problems can be stated in a single common form: One is given a set of quantizers described by their distortion–rate performances as

$$D_i = g_i(R_i), \quad R_i \in \mathcal{R}_i, \quad i = 1, 2, \dots, N.$$

Each set of available rates  $\mathcal{R}_i$  is a subset of the nonnegative real numbers and may be discrete or continuous. The problem is to

$$\begin{aligned} &\text{minimize average distortion} & D &= N^{-1} \sum_{n=0}^{N-1} D_n, \\ &\text{subject to maximum average rate} & R &= N^{-1} \sum_{n=0}^{N-1} R_n. \end{aligned}$$

If the average distortion can be reduced by taking bits away from one component and giving them to another, the initial bit allocation is not optimal. Applying this reasoning with infinitesimal changes in the component rates, a necessary condition for an optimal allocation is that the slope of each  $g_i$  at  $R_i$  is equal to a common constant value.

The approximate performance given by (5.101) leads to a particularly easy bit allocation problem with

$$g_i = c_i \sigma_i^2 2^{-2R_i}, \quad \mathcal{R}_i = [0, \infty), \quad i = 1, 2, \dots, N. \quad (5.104)$$

Ignoring the fact that each component rate must be nonnegative, an equal-slope argument shows that the optimal bit allocation is

$$R_i = R + \frac{1}{2} \log_2 \frac{c_i}{\left(\prod_{i=1}^N c_i\right)^{1/N}} + \frac{1}{2} \log_2 \frac{\sigma_i^2}{\left(\prod_{i=1}^N \sigma_i^2\right)^{1/N}}.$$

With these rates, all the  $D_i$ 's are equal and the average distortion is

$$D = \left(\prod_{i=1}^N c_i\right)^{1/N} \left(\prod_{i=1}^N \sigma_i^2\right)^{1/N} 2^{-2R}. \quad (5.105)$$

This solution is valid when each  $R_i$  given above is nonnegative. For lower rates, the components with smallest  $c_i \sigma_i^2$  are allocated no bits and the remaining components have correspondingly higher allocations.

**Bit Allocation with Uniform Quantizers** With uniform quantizers, bit allocation is nothing more than choosing a step size  $\Delta_i$  for each of the  $N$  components. The equal-distortion property of the analytical bit allocation solution gives a simple rule: Make all of the step sizes equal. This will be referred to as *lazy* bit allocation. Our development indicates that lazy allocation is optimal when the rate is high. Numerical studies have shown that lazy allocation is nearly optimal as long as the minimal allocated rate is at least 1 bit. Entropy-constrained uniform quantization with lazy bit allocation is used in the numerical examples in the following section.

### 5.5.2 Optimal Transforms for Transform Coding

We are now ready for the main event of designing the analysis transform  $\tilde{\Phi}$  and the synthesis transform  $\Phi$ . Throughout this section the source  $\mathbf{x}$  is assumed to have mean zero, and  $\Sigma_{\mathbf{x}}$  denotes the covariance matrix  $E[\mathbf{x}\mathbf{x}^T]$ . The source is often, but not always, jointly Gaussian.

A signal given as a vector in  $\mathbb{R}^N$  is implicitly represented as a series with respect to the standard basis. An invertible analysis transform  $\tilde{\Phi}$  changes the basis. A change of basis does not alter the information in a signal, so how can it affect coding efficiency? Indeed, if arbitrary source coding is allowed after the transform, it does not. The motivating principle of transform coding is that *simple* coding may be more effective in the transform domain than in the original signal space. In the standard model, simple coding corresponds to the use of scalar quantization and scalar entropy coding.

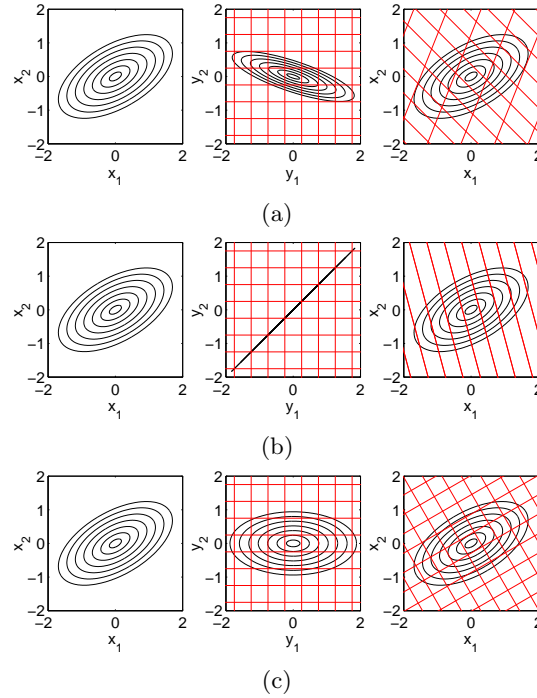
**Visualizing Transforms** Beyond two or three dimensions, it is difficult to visualize vectors—let alone the action of a transform on vectors. Fortunately, we already have an idea of what a linear transform does: it combines rotating, scaling, and shearing such that a hypercube is always mapped to a parallelepiped. For example, in two dimensions, the level curves of a zero-mean Gaussian density are ellipses centered at the origin with collinear major axes, as shown in the left panels of Figure 5.30.<sup>96</sup> The middle panel of Figure 5.30(a) shows the level curves of the joint density of the expansion coefficients after an arbitrary invertible linear transformation. A linear transformation of an ellipse is still an ellipse, though its eccentricity and orientation (direction of major axis) may have changed.

The grid in the middle panel indicates the cell boundaries in uniform scalar quantization, with equal step sizes, of the transform coefficients. The effect of inverting the transform is shown in the right panel; the source density is returned to its original form and the quantization partition is linearly deformed. The partition in the original coordinates, as shown in the right panel, is what is truly relevant. It shows which source vectors are mapped to the same symbol, thus giving some indication of the average distortion. Looking at the number of cells with appreciable probability gives some indication of the rate.

A singular transform is a degenerate case. As shown in the middle panel of Figure 5.30(b), the transform coefficients have probability mass only along a line. (A line segment is an ellipse with unit eccentricity.) Inverting the transform is not possible, but we may still return to the original coordinates to view the partition induced by quantizing the expansion coefficients. The cells are unbounded in one direction, as shown in the right panel. This is undesirable unless variation of the source in the direction in which the cells are unbounded is very small.

Although better than unbounded cells, the parallelogram-shaped partition cells that arise from arbitrary invertible transforms are inherently suboptimal (see Example 5.18). To get rectangular partition cells, the basis vectors must be orthogonal, shown in Figure 5.30(c) for the KLT. For square cells, when quantization step

<sup>96</sup>Because the covariance matrix is symmetric, it has an orthogonal set of eigenvectors, and thus orthogonal principal axes.



**Figure 5.30:** Illustration of various basis changes. The Gaussian source is depicted by level curves of the PDF (left). The expansion coefficients are separately quantized with uniform quantizers (center). The induced partitioning is then shown in the original coordinates (right). (a) A basis change generally induces a non-hypercubic partition. (b) A singular transformation gives a partition with unbounded cells. (c) A Karhunen–Loève transform aligns the partitioning with the axes of the source PDF.

sizes are equal for each transform coefficient, the basis vectors should in addition to being orthogonal have equal lengths.

**EXAMPLE 5.18 (SHAPES OF PARTITION CELLS)** The quality of a source code depends on the shapes of the partition cells  $\{\tilde{\beta}^{-1}(i), i \in \mathcal{I}\}$  and on varying the sizes of the cells according to the source density. When the rate is high, and either the source is uniformly distributed or the rate is measured by entropy ( $H(\tilde{\beta}(\mathbf{x}))$ ), the sizes of the cells should essentially not vary. Then, the quality depends on having cell shapes that minimize the average distance to the center of the cell.

For a given volume, a body in Euclidean space that minimizes the average distance to the center is a sphere. But spheres do not work as partition cell shapes because they do not pack together without leaving interstices. Only for a few dimensions  $N$  is the best cell shape known. One such dimension is  $N = 2$ , where the hexagonal packing shown in Figure 5.7(b) is best.

The best packings (including the hexagonal case) cannot be achieved with transform codes. Transform codes can only produce partitions into parallelepipeds,

as shown for  $N = 2$  in Figure 5.30. The best parallelepipeds are cubes. We get a hint of this by comparing a rectangular partitions of a unit-area square as well as the square one shown in Figure 5.7(a). Both partitions have 36 cells, so each cell has the same area. The partition with square cells gives distortion  $1/432 \approx 2.31 \times 10^{-3}$ , while the other gives  $97/31104 \approx 3.12 \times 10^{-3}$ . (The calculations are easy; see our discussion on uniform quantization.)

This simple example can also be interpreted as a problem of allocating bits between the horizontal and vertical components. The lazy bit allocation arising from equal quantization step sizes for each component is optimal. This holds generally for high-rate entropy-constrained quantization of components with the same normalized density.

**Orthonormal Transforms with Gaussian Sources** Consider a jointly Gaussian source, and assume  $\tilde{\Phi} = \Phi^T$ , that is, the transform is orthonormal. The Gaussian assumption is important because any linear combination of jointly Gaussian random variables is Gaussian. Thus, any analysis transform gives Gaussian expansion coefficients. Then, since the expansion coefficients have the same normalized density, for any reasonable set of quantizers, (5.96) holds with a single function  $g(R)$  describing all of the expansion coefficients. Orthogonality is important because orthogonal transforms preserve Euclidean lengths, which gives  $d(x, \hat{x}) = d(y, \hat{y})$ .

With these assumptions, for any rate and bit allocation a KLT is an optimal transform:

**THEOREM 5.9** Consider a transform coder with orthogonal analysis/synthesis transforms,  $\tilde{\Phi} = \Phi^T$ . Suppose there is a single function  $g$  to describe the quantization of each expansion coefficient through

$$\mathbb{E}[(\alpha_i - \hat{\alpha}_i)^2] = \sigma_i^2 g(R_i), \quad i = 1, 2, \dots, N,$$

where  $\sigma_i^2$  is the variance of  $\alpha_i$  and  $R_i$  is the rate allocated to  $\alpha_i$ . Then, for any bit allocation  $(R_1, R_2, \dots, R_N)$ , there exists a KLT that minimizes the distortion. In the typical case where  $g$  is nonincreasing, a KLT that gives  $(\sigma_1^2, \sigma_2^2, \dots, \sigma_N^2)$  sorted in the same order as the bit allocation minimizes the distortion.

Recall that with a high average rate of  $R$  bits per component and quantizer performance described by (5.104), the average distortion with optimal bit allocation is given by (5.105). With Gaussian expansion coefficients that are optimally quantized, the distortion simplifies to

$$D = c \left( \prod_{i=1}^N \sigma_i^2 \right)^{1/N} 2^{-2R}, \quad (5.106)$$

where  $c = (1/6)\pi e$  for entropy-constrained quantization or  $c = (1/2)\sqrt{3}\pi$  for fixed-rate quantization. The choice of an orthogonal transform is thus guided by minimizing the geometric mean of the expansion coefficient variances.



**THEOREM 5.10** The distortion given by (5.106) is minimized over all orthogonal transforms by any KLT.

*Proof.* Applying Hadamard's inequality that states that the absolute value of the determinant of a matrix is bounded from above by the product of the norms of the column vectors to  $\Sigma_\alpha$  gives

$$(\det \tilde{\Phi})(\det \Sigma_x)(\det \tilde{\Phi}^T) = \det \Sigma_\alpha \leq \prod_{i=1}^N \sigma_i^2.$$

Since  $\det \tilde{\Phi} = 1$ , the left-hand side of this inequality is invariant under the choice of  $\tilde{\Phi}$ . Equality is achieved when a KLT is used. Thus KLT minimizes the distortion.

Equation (5.106) can be used to define a figure of merit called the coding gain. The *coding gain* of a transform is a function of its variance vector,  $[\sigma_1^2 \ \sigma_2^2 \ \dots \ \sigma_N^2]$ , and the variance vector without a transform,  $\text{diag}(\Sigma_x)$ :

$$\text{coding gain} = \frac{\left(\prod_{n=0}^{N-1} (\Sigma_x)_{nn}\right)^{1/N}}{\left(\prod_{n=0}^{N-1} \sigma_n^2\right)^{1/N}}.$$

The coding gain is the factor by which the distortion is reduced because of the transform, assuming high rate and optimal bit allocation. The foregoing discussion shows that KLTs maximize coding gain.

## 5.6 Computational Aspects

Many algorithms are associated to sampling, interpolation, quantization, and approximation. We discuss a few representative examples.

### 5.6.1 Optimal Quantization and Clustering

In Section 5.5.1, we saw simple, uniform quantization. A better way to do quantization uses nonuniform intervals when the PDF of the random variable(s) is nonuniform.

**EXAMPLE 5.19 (LLOYD'S ALGORITHM)**

One reason for the popularity of DCT is the existence of a fast,  $N \log N$  algorithm for its computation. Because the DCT is a trigonometric transform, and resembles a real version of a DFT, it is not surprising that one can use the FFT to compute the DCT.

### 5.6.2 Projection Onto Convex Sets

Another example combines bandlimitedness and quantization using POCS.

**EXAMPLE 5.20 (OVERSAMPLING AND QUANTIZATION)**

## Chapter at a Glance

### Approximation of Functions by Polynomials

Method	Approximation criterion	Approximating polynomial $p_K(t)$	Error $\epsilon_K(t)$
Least-squares	$\min_{p_K} \ x - p_K\ _2^2$	$\sum_{k=0}^K \langle x, \varphi_k \rangle \varphi_k(t)$	$x(t) - p_K(t)$
Lagrange	$p_K(t_k) = x(t_k)$	$\sum_{k=0}^K x(t_k) \prod_{\substack{i=0 \\ i \neq k}}^K \frac{t - t_i}{t_k - t_i}$	$\frac{\prod_{k=0}^K (t - t_k)}{(K+1)!} x^{(K+1)}(\xi)$
Taylor series	$p_K^{(k)}(t_0) = x^{(k)}(t_0)$	$\sum_{k=0}^K \frac{(t - t_0)^k}{k!} x^{(k)}(t_0)$	$\frac{(t - t_0)^{K+1}}{(K+1)!} x^{(K+1)}(\xi)$
Minimax	$\min_{p_K} \ x - p_K\ _\infty$		

Polynomials	Definition	Recursion	Weight	Interval
Legendre $L_k(t)$	$\frac{(-1)^k}{2^k k!} \frac{d^k}{dt^k} (1 - t^2)^k$	$\frac{2k-1}{k} t L_{k-1} - \frac{k-1}{k} L_{k-2}$	1	$[-1, 1]$
Chebyshev $T_k(t)$	$\cos(k \arccos t)$	$2t T_{k-1} - T_{k-2}$	$\frac{1}{\sqrt{1-t^2}}$	$[-1, 1]$
Laguerre $L_k(t)$	$\frac{e^t}{k!} \frac{d^k}{dt^k} (e^{-t} t^k)$	$\frac{2k-1-t}{k} L_{k-1} - \frac{k-1}{k} L_{k-2}$	$e^{-t}$	$[0, \infty)$
Hermite $H_k^{(a)}(t)$	$a^{-3k/2} k! t^k \cdot \sum_{\ell=0}^{\lfloor k/2 \rfloor} \frac{(-2t^2/a)^{-\ell}}{\ell!(k-2\ell)!}$	$2t H_{k-1} - 2(k-1)H_{k-2}$	$e^{-t^2}$	$(-\infty, \infty)$

### Approximation of Functions by Splines

Method	Approximation
B-splines	$\hat{x}(t) = \sum_{k \in \mathbb{Z}} \langle x(t), \tilde{\beta}^{(N)}(t-k) \rangle_t \beta^{(N)}(t-k)$ $\beta^{(0)}(t) = \begin{cases} 1, &  t  \leq 1/2; \\ 0, & \text{otherwise,} \end{cases} \quad \xleftrightarrow{\text{FT}} \quad B^{(0)}(\omega) = \text{sinc}\left(\frac{\omega}{2}\right)$ $\beta^{(N)}(t) = \beta^{(N-1)}(t) * \beta^{(0)}(t), \quad \xleftrightarrow{\text{FT}} \quad B^{(N)}(\omega) = \left(\text{sinc}\left(\frac{\omega}{2}\right)\right)^{N+1}$
Orthogonalized splines	$\hat{x}(t) = \sum_{k \in \mathbb{Z}} \langle x(t), \varphi^{(N)}(t-k) \rangle_t \varphi^{(N)}(t-k)$ $\varphi^{(N)}(t) = \sum_{k=0}^{\infty} d_k^{(N)} \beta^{(N)}(t-k)$
Polynomial reproduction	$p_N(t) = \sum_{k \in \mathbb{Z}} \alpha_k \beta^{(N)}(t-k)$ $\sum_{n \in \mathbb{Z}} \varphi(t-n) = 1 \quad \xleftrightarrow{\text{FT}} \quad \Phi(2\pi k) = \delta_k$

### Approximation of Functions and Sequences by Series Truncation

Method	Approximation $\hat{x}_M(t)$	Coefficients used	Error $\epsilon_M^2$
Linear	$\sum_{k=0}^{M-1} \langle x, \varphi_k \rangle \varphi_k$	First $M$	$\sum_{k=M}^{\infty}  \langle x, \varphi_k \rangle ^2$
Nonlinear	$\sum_{k \in \mathcal{I}_M} \langle x, \varphi_k \rangle \varphi_k$	Largest $M$	$\sum_{k \notin \mathcal{I}_M}  \langle x, \varphi_k \rangle ^2$

### Historical Remarks



One of the names appearing prominently in this chapter is that of **Pafnuty Lvovich Chebyshev (1821-1894)**, considered to be the founding father of Russian mathematics. His contributions are many, in fields ranging from probability and statistics, to number theory. Chebyshev polynomials were described in this chapter for use in minimax approximation; they are also responsible for Chebyshev's name finding its way into signal processing, through the family of Chebyshev filters. As an interesting aside, a crater on the Moon was named after Chebyshev.

The origins of splines are particularly interesting. They date back to ship building; naval engineers needed a method to thread a smooth curve through a given set of points. This resulted in thin wooden strips, *splines*, placed between pairs of points, *ducks*, *rats*, or *dogs*. The method was then used in both the aircraft as well as the automobile industry in the late 1950s and early 1960s. Engineers at Citroën, Renault and General Motors developed the theory further; in particular, Pierre Bézier, a French engineer working at Renault, became a leader in using mathematical and computational tools in design and manufacturing. With the advent of computers, splines took over from polynomials as a tool for interpolating functions.

On the compression side, transform coding was invented as a method for conserving bandwidth in the transmission of signals output by the analysis unit of a ten-channel vocoder (*voice coder*) [49]. These correlated, continuous-time, continuous-amplitude signals represented estimates, local in time, of the power in ten contiguous frequency bands. By adding modulated versions of these power signals, the synthesis unit resynthesized speech. The vocoder's ancestor, *Pedro*, the *Voder*, was presented at the 1939 World's Fair. Kramer and Mathews [92] showed that the total bandwidth necessary to transmit the signals with a prescribed fidelity can be reduced by transmitting an appropriate set of linear combinations of the signals instead of the signals themselves. This is not source coding because it does not involve discretization. Thus, one could ascribe a later birth to transform coding. Huang and Schultheiss [77] introduced the structure we called the standard model (bottom of Figure 5.27). They studied the coding of Gaussian sources while assuming independent expansion coefficients and optimal fixed-rate scalar quantization. They first showed that  $U = T^{-1}$  is optimal and then that  $T$  should have orthogonal rows. Transform coding has since spread into almost all aspects of our lives, through its use in the popular media standards, such as MP3, JPEG and MPEG.



### Further Reading

An excellent introductory text to numerical analysis is by Atkinson [6]. For splines, the magazine review article by Unser [154], gives a thorough overview of splines and their use in signal processing, together with a number of references. The book by Strang and Fix, [142], contains further details on polynomial reproduction and Strang–Fix Theorem.

For compression and transform coding, the review by Goyal, [61], is the basis of Section 5.5, and contains further details and generalizations, as do [33, 58, 60, 64]. Details on *high-resolution* quantization theory for both scalars and vectors can be found in [57, 64] and references therein.

### Exercises with Solutions

#### 5.1. Chebyshev Polynomials

- (i) Using the trigonometric identity

$$\cos((k \pm 1)\theta) = \cos(k\theta)\cos(\theta) \mp \sin(k\theta)\sin(\theta), \quad (\text{E5.1-1})$$

prove the recursion (5.17) for Chebyshev polynomials.

- (ii) Using (5.17), prove that the Chebyshev polynomials are polynomial functions in  $t$ .  
 (iii) Show that under the inner product (5.16), the Chebyshev polynomials satisfy

$$\langle T_n, T_m \rangle = \begin{cases} 0, & \text{for } n \neq m; \\ \pi, & \text{for } n = m = 0; \\ \pi/2, & \text{for } n = m > 0. \end{cases}$$

(Hint: The change of variables  $\theta = \cos^{-1} t$  simplifies the integrals.)

- (iv) Show that the leading coefficient in the Chebyshev polynomial  $T_k(t)$ ,  $k \in \mathbb{Z}^+$ , is  $2^{k-1}$ .  
 (v) Prove the expressions for the zeros, (5.18), as well as extrema, (5.19), of Chebyshev polynomials.

*Solution:*

- (i) With  $T_k(t) = \cos(k \arccos t)$  as in (5.15), and  $\theta = \arccos t$ ,

$$\begin{aligned} T_{k+1}(t) &= \cos((k+1)\theta) \stackrel{(a)}{=} \cos(k\theta)\cos(\theta) - \sin(k\theta)\sin(\theta) \\ &= \cos(k\theta)\cos(\theta) - \sin(k\theta)\sin(\theta) \pm \cos(k\theta)\cos(\theta) \\ &= 2\cos(k\theta)\cos(\theta) - (\cos(k\theta)\cos(\theta) + \sin(k\theta)\sin(\theta)) \\ &\stackrel{(b)}{=} 2\cos(k\theta)\cos(\theta) - \cos((k-1)\theta) = 2tT_k(t) - T_{k-1}(t), \end{aligned}$$

where both (a) and (b) follow from (E5.1-1).

- (ii) We can prove this by inductions. First,

$$T_2(t) = 2tT_1(t) - T_0(t) = 2t^2 - 1,$$

a polynomial function. In the induction step, if  $T_k(t)$  and  $T_{k-1}(t)$  are polynomial functions in  $t$ , then

$$\begin{aligned} T_{k+1}(t) &= 2tT_k(t) - T_{k-1}(t) = 2t \sum_{n=0}^k \alpha_n t^n - \sum_{n=0}^{k-1} \beta_n t^n \\ &= \sum_{n=0}^k 2\alpha_n t^{n+1} - \sum_{n=0}^{k-1} \beta_n t^n = \sum_{n=1}^{k+1} 2\alpha_{n-1} t^n - \sum_{n=0}^{k-1} \beta_n t^n \\ &= -\beta_0 + \sum_{n=1}^{k-1} (2\alpha_{n-1} - \beta_n) t^n + 2\alpha_{k-1} t^k + 2\alpha_k t^{k+1} = \sum_{n=0}^{k+1} \gamma_n t^n, \end{aligned}$$

clearly a polynomial function in  $t$ .

(iii) Using (5.16), we have

$$\begin{aligned}\langle T_n, T_m \rangle &= \int_{-1}^1 T_n(t) T_m(t) (1-t^2)^{-1/2} dt \\ &= \int_{-1}^1 \cos(n \arccos t) \cos(m \arccos t) (1-t^2)^{-1/2} dt.\end{aligned}$$

We first solve for  $n = m = 0$ :

$$\langle T_n, T_m \rangle = \int_{-1}^1 (1-t^2)^{-1/2} dt = \arcsin t \Big|_{-1}^1 = (2k+1)\frac{\pi}{2} - (2k+3)\frac{\pi}{2} = -\pi,$$

which is the same as  $\pi$ .

Next, we solve for  $n = m > 0$ :

$$\begin{aligned}\langle T_n, T_m \rangle &= \int_{-1}^1 \cos(n \arccos t)^2 (1-t^2)^{-1/2} dt \\ &= -\frac{1}{2} \arccos t \Big|_{-1}^1 + \frac{1}{4n} \sin(2n \arccos t) \Big|_{-1}^1 = -\frac{1}{2}(2k\pi - (2k+1)\pi) = \frac{\pi}{2}.\end{aligned}$$

Finally, we solve for  $n \neq m$ :

$$\begin{aligned}\langle T_n, T_m \rangle &= \int_{-1}^1 \cos(n \arccos t) \cos(m \arccos t) (1-t^2)^{-1/2} dt \\ &= \frac{1}{2} \int_{-1}^1 (\cos((n+m) \arccos t) + \cos((n-m) \arccos t)) (1-t^2)^{-1/2} dt \\ &= \frac{1}{2} \left( \frac{\sin((n+m) \arccos t)}{n+m} \Big|_{-1}^1 + \frac{\sin((n-m) \arccos t)}{n-m} \Big|_{-1}^1 \right) = 0.\end{aligned}$$

(iv) We can again use induction to prove this. For  $T_1(t) = t$ , the leading coefficient is  $2^0$ . Assuming that the leading coefficient for  $T_k(t)$  is  $2^{k-1}$ , then using the result of (ii), we see that the leading coefficient of  $T_{k+1}(t)$  is

$$\beta_{k+1} = 2\alpha_k = 2 \cdot 2^{k-1} = 2^k.$$

(v) From the expression for  $T_k(t) = \cos(k \arccos t)$ , we get that the zeros are at

$$(2m+1)\frac{\pi}{2} = k \arccos t, \quad \Rightarrow \quad t_m = \cos\left(\frac{2m+1}{2k}\pi\right),$$

for  $m = 0, 1, \dots, k-1$ . Similarly, the extrema are at

$$m\pi = k \arccos t, \quad \Rightarrow \quad t_m = \cos\left(\frac{m}{k}\pi\right),$$

for  $m = 0, 1, \dots, k$ .

### 5.2. Strang-Fix Theorem for an Interpolating $\varphi(t)$

Assume an interpolating function  $\varphi(t)$  as in (5.70) that is sufficiently localized as in (5.68). Prove that (i) and (ii) in Theorem 5.7 are equivalent.

*Solution:* Condition (i) means that there exist coefficient sequences  $\alpha_n^{(k)}$  such that

$$\sum_{n \in \mathbb{Z}} \alpha_n^{(k)} \varphi(t-n) = (t-t_0)^k, \quad k = 0, 1, \dots, K. \quad (\text{E5.2-1})$$

Because of the interpolation property, when  $t = m$ ,

$$\sum_{n \in \mathbb{Z}} \alpha_n^{(k)} \varphi(m-n) = \alpha_m^{(k)} = (m-t_0)^k.$$

Thus, (E5.2-1) becomes

$$\sum_{n \in \mathbb{Z}} (n - t_0)^k \varphi(t - n) = (t - t_0)^k \quad k = 0, 1, \dots, K,$$

which, by setting  $t_0 = t$ , yields

$$\sum_{n \in \mathbb{Z}} (n - t)^k \varphi(t - n) = \begin{cases} 1, & k = 0; \\ 0, & k = 1, 2, \dots, K \end{cases} \quad (\text{E5.2-2})$$

The left-hand side is a 1-periodic function, and because of localization (5.68), converges to an  $\mathcal{L}^1$  function on the  $[0, 1]$  interval. This function has a Fourier series representation for each  $k = 0, 1, \dots, K$ . The coefficients are

$$\begin{aligned} c_\ell^{(k)} &= \int_0^1 \sum_{n \in \mathbb{Z}} (n - t)^k \varphi(t - n) e^{-j2\pi\ell t} dt \stackrel{(a)}{=} \sum_{n \in \mathbb{Z}} \int_0^1 (n - t)^k \varphi(t - n) e^{-j2\pi\ell t} dt \\ &\stackrel{(b)}{=} \int_{-\infty}^{\infty} (-t)^k \varphi(t) e^{-j2\pi\ell t} dt \stackrel{(c)}{=} \frac{1}{(2\pi j)^k} \Phi^{(k)}(2\pi\ell), \end{aligned} \quad (\text{E5.2-3})$$

where (a) follows from (1.196); in (b) we merge individual integrals into a single integral over the real line; and (c) follows from the Fourier transform, (3.48a), as well as the differentiation property in frequency in Table 3.2. For  $k = 0$ , the function in (E5.2-2) is equal to 1, so the Fourier series coefficients are  $c_\ell^{(0)} = \delta_\ell$  and thus,

$$\Phi(2\pi\ell) = \delta_\ell \quad \ell \in \mathbb{Z}.$$

For  $k = 1, 2, \dots, K$ , the function in (E5.2-2) is zero, or

$$c_\ell^{(k)} = 0 \quad k = 1, 2, \dots, K$$

leading to

$$\Phi^{(k)}(2\pi\ell) = 0 \quad k = 1, 2, \dots, K, \quad \ell \in \mathbb{Z},$$

hence verifying (5.69).

### 5.3. Sampling Discrete-Time Periodic Stream of Kronecker Delta Pulses

Let  $x_n$  be a discrete-time periodic signal of period  $N$ , containing  $K$  weighted Kronecker delta pulses at locations  $\{n_0, n_1, \dots, n_{K-1}\}$ ,  $n_\ell \in [0, N-1]$  and  $K < \lfloor N/2 \rfloor$ ,

$$x_n = \sum_{\ell=0}^{K-1} c_\ell \delta_{n-n_\ell},$$

and let  $X_k$  be its DFT from (2.159a).

The sequence  $x_n$  is filtered with the time-reversed version of an ideal lowpass filter  $h_n = (1/N) \sum_{k=-K}^K W_N^{-kn}$  and downsampled by  $M$ ,

$$y_\ell = (h_{\ell M - n} \otimes x_n)_n, \quad \ell = 0, 1, \dots, \frac{N}{M} - 1,$$

where  $M$  is an integer divisor of  $N$  satisfying  $N/M \geq 2K + 1$ .

- (i) Prove that the DFT coefficients  $X_k$ ,  $k \in [-K, K]$  are sufficient to determine the locations of  $K$  weighted Kronecker delta pulses.
- (ii) Prove that the  $N/M$  samples  $y_\ell$  are a sufficient representation of  $X_k$ ,  $k \in [-K, K]$ , that is, find the relation between  $X_k$  and  $Y_k$ .

*Solution:*

- (i) The DFT coefficients are given by

$$X_k = \sum_{\ell=0}^{K-1} x_\ell W_N^{mn_\ell}, \quad \ell = 0, 1, \dots, K-1. \quad (\text{E5.3-1})$$

Since  $X_k$  is a linear combination of  $K$  complex exponentials,  $W_N^{n_k}$ , the locations  $n_k$  of the Kronecker delta pulses can be found using the following method: Find the so-called *annihilating filter*

$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_K z^{-K} = \prod_{k=0}^{K-1} (1 - W_N^{n_k} z^{-1}),$$

such that

$$\sum_{k=0}^K a_k X_{m-k} = 0, \quad m = 0, 1, \dots, N-1. \quad (\text{E5.3-2a})$$

If we can find the unknown coefficients  $a_k$ ,  $k = 1, 2, \dots, K$ , we will find the unknown Kronecker delta pulse locations  $n_\ell$ ,  $\ell = 1, 2, \dots, K$ , because  $W_N^{n_k}$  are the roots of  $A(z)$ . Since  $a_0 = 1$ ,  $K$  equations (E5.3-2a) are sufficient to determine the  $K$  unknown filter coefficients  $a_k$ . Let  $m = 1, 2, \dots, K$ , then the system in (E5.3-2a) is equivalent to

$$\sum_{k=1}^K a_k X_{m-k} = -X_m, \quad m = 1, 2, \dots, K.$$

Writing this in matrix form,

$$\begin{bmatrix} X_0 & X_{-1} & \cdots & X_{-K+1} \\ X_1 & X_0 & \cdots & X_{-K+2} \\ \vdots & \vdots & \ddots & \vdots \\ X_K & X_{K-1} & \cdots & X_0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_K \end{bmatrix} = - \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_K \end{bmatrix}. \quad (\text{E5.3-2b})$$

Because  $X_k$  are linear combinations of complex exponentials, the matrix in (E5.3-2b) is of full rank  $K$ . Thus, there exists a unique solution  $\{a_1, a_2, \dots, a_K\}$ . The set of locations  $\{n_0, n_1, \dots, n_{K-1}\}$  can then be found as the zeros of  $A(z)$ .

The weights of the Kronecker delta pulses are obtained by solving the  $K$  DFT equations (E5.3-1), leading to a Vandermonde system

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \\ W_N^{n_0} & W_N^{n_1} & \cdots & W_N^{n_{K-1}} \\ \vdots & \vdots & \ddots & \vdots \\ W_N^{n_0(K-1)} & W_N^{n_1(K-1)} & \cdots & W_N^{n_{K-1}(K-1)} \end{bmatrix} \begin{bmatrix} x_{n_0} \\ x_{n_1} \\ \vdots \\ c_{n_{K-1}} \end{bmatrix} = \begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{K-1} \end{bmatrix},$$

and has a unique solution since the locations are distinct (see (1.231)).

(ii) For each  $\ell = 0, 1, \dots, N/M - 1$ ,

$$\begin{aligned} y_\ell &= (h_{\ell M - n} \otimes x_n)_n = \sum_{n=0}^{N-1} x_n h_{n - \ell M} \\ &\stackrel{(a)}{=} \frac{1}{N} \sum_{n=0}^{N-1} x_n \sum_{m=-K}^K W_N^{-m(n - \ell M)} = \frac{1}{N} \sum_{n=0}^{N-1} x_n \sum_{m=-K}^K W_N^{-mn} W_N^{m\ell M} \\ &\stackrel{(b)}{=} \frac{1}{N} \sum_{m=-K}^K W_{N/M}^{m\ell} \sum_{n=0}^{N-1} x_n W_N^{-nm} = \frac{1}{N} \sum_{m=-K}^K X_{-m} W_{N/M}^{m\ell}, \quad (\text{E5.3-3}) \end{aligned}$$

where (a) follows from taking the DFT of  $h_n$ ; in (b) we exchanged the order of summations and used  $W_N^{m\ell M} = W_{N/M}^{m\ell}$ ; and in (c) we used the DFT of the second summation.

We now find the DFT of  $y_\ell$ . For each  $k = 0, 1, \dots, N/M - 1$ ,

$$\begin{aligned} Y_k &= \sum_{\ell=0}^{N/M-1} y_\ell W_{N/M}^{\ell k} \stackrel{(a)}{=} \frac{1}{N} \sum_{\ell=0}^{N/M-1} \sum_{m=-K}^K X_{-m} W_{N/M}^{m\ell} W_{N/M}^{\ell k} \\ &\stackrel{(b)}{=} \frac{1}{N} \sum_{m=-K}^K X_{-m} \sum_{\ell=0}^{N/M-1} W_{N/M}^{\ell(k+m)} \stackrel{(c)}{=} \frac{1}{M} X_k, \end{aligned}$$

where (a) follows from (E5.3-3); in (b) we exchanged the order of summations and multiplied the two complex exponentials; (c) follows from the orthogonality of roots of unity, (2.277c), making the second sum go to zero except for  $m = -k$  when it equals  $N/M$ , where  $k = 0, 1, \dots, \min\{K, N/M - 1\}$ .

By assumption,  $N/M \geq 2K + 1 > K$ , and thus

$$X_k = M Y_k, \quad k = 0, 1, \dots, K.$$

#### 5.4. Quantization Intervals

Let  $x$  be a real-valued random variable with PDF  $f_x$ . This random variable is quantized into  $K$  representation values  $\hat{x}_k$ ,  $k = 0, 1, \dots, K - 1$ , with  $\hat{x}_k < \hat{x}_{k+1}$  for all  $k$ . Show that the nearest neighbor assignment,

$$x \rightarrow \hat{x}_k \quad \text{when } |x - \hat{x}_k| \leq |x - \hat{x}_i| \text{ for all } i \neq k,$$

minimizes the mean-squared error  $E[(x - \hat{x})^2]$ . This leads to a midpoint splitting rule, where values  $x \in [\hat{x}_k, \hat{x}_{k+1}]$  are assigned to  $\hat{x}_k$  if smaller than  $(\hat{x}_k + \hat{x}_{k+1})/2$  and to  $\hat{x}_{k+1}$  otherwise. (In multiple dimensions, this leads to Voronoi cells.)

*Solution:* Denote the quantization mapping by  $q: \mathbb{R} \rightarrow \{\hat{x}_k\}_{k=0}^{K-1}$  so that  $\hat{x} = q(x)$ . Since

$$E[(x - \hat{x})^2] = \int_{-\infty}^{\infty} (t - q(t))^2 f_x(t) dt$$

and  $f_x$  is nonnegative, the mean-squared error is minimized by minimizing  $(t - q(t))^2$  at each point  $t$ . For any  $t \in \mathbb{R}$ , out of the choices for  $q(t)$  in  $\{\hat{x}_k\}_{k=0}^{K-1}$ , the squared error  $(t - q(t))^2$  is minimized by choosing  $q(t)$  to be the representation value  $\hat{x}_k$  closest to  $t$ . Specifically for  $t \in [\hat{x}_k, \hat{x}_{k+1}]$ , the distance from  $\hat{x}_k$  is an increasing function and the distance from  $\hat{x}_{k+1}$  is a decreasing function; these distances are equal at the midpoint, where  $q(t)$  changes from  $\hat{x}_k$  to  $\hat{x}_{k+1}$ .

## Exercises

### 5.1. Basic Properties of Legendre Polynomials

- (i) Let  $V = \{v_0, v_1, \dots\}$  be the set of polynomials orthogonal on  $\mathcal{L}^2([a, b])$ ; each  $v_k$  has degree at most  $k$ . Express  $V$  in terms of Legendre polynomials.
- (ii) Prove the following recurrence relation for Legendre polynomials:

$$L_{k+1}(t) = \frac{2k+1}{k+1} t L_k(t) - \frac{k}{k+1} L_{k-1}(t) \quad k \in \mathbb{Z}^+.$$

### 5.2. Orthogonal Polynomials and Nesting of Polynomial Subspaces

Let  $V = \{v_0, v_1, \dots\}$  be a set of orthogonal polynomials; each  $v_k$  has degree at most  $k$ . Let  $p$  be a polynomial of degree  $m$ . Prove that  $\langle p, v_k \rangle = 0$  for every  $k > m$ .

### 5.3. Roots of Orthogonal Polynomials

Let  $V = \{v_0, v_1, \dots\}$  be the set of polynomials orthogonal on  $[a, b]$ ; each  $v_k$  has degree at most  $k$ . Prove that for any  $k$ ,  $v_k(t)$  has exactly  $k$  real, distinct roots in the open interval  $(a, b)$ .

(Hint: [TBD, Atkinson p. 213–214].)

### 5.4. Poor $\mathcal{L}^\infty$ Behavior of Lagrange Interpolation

Let  $x(t) = (1 + t^2)^{-1}$  and let  $p_K(t)$  denote the Lagrange interpolation of  $K + 1$  samples of  $x$  evenly-spaced over  $[-5, 5]$ .

- (i) Bound  $|x(t) - p_K(t)|$  over  $[-5, 5]$  by evaluating (5.7b).
- (ii) Show that the  $\mathcal{L}^\infty$  error bound from (i) grows without bound as  $K$  is increased.



- (iii) In Matlab, plot  $p_K$  for a few  $K$ s. Observe empirically that  $\|x - p_K\|_\infty$  grows without bound as  $K$  is increased.

5.5. *Lagrange Interpolation with Coincident Nodes*

Given is the Lagrange interpolation formula (5.5).

- (i) Write it for two nodes, one at 0 and the other at  $\epsilon > 0$ . Using the definition of the derivative, prove that in the limit, when  $\epsilon \rightarrow 0$ , the Lagrange interpolation yields the first-order Taylor series expansion around 0.
- (ii) Generalize (i) to arbitrary order.

5.6. *Poor  $\mathcal{L}^\infty$  Behavior of Taylor Series*

Let  $x(t) = (1 + t^2)^{-1}$  and let  $p_K(t)$  denote the Taylor series approximation over  $[-5, 5]$ . This function is infinitely differentiable but potentially difficult to approximate with a polynomial [6].

- (i) Bound  $|x(t) - p_K(t)|$  over  $[-5, 5]$ , with Taylor series approximating around 0.
- (ii) Bound  $|x(t) - p_K(t)|$  over  $[-5, 5]$  by evaluating (5.10b) and compare it to (i).
- (iii) Show that the  $\mathcal{L}^\infty$  error bounds from (i) and (ii) grow without bound as  $K$  is increased.

5.7. *Hermite Interpolation*

Let  $x^{(i)}(t_k)$  be the values of a real-valued function and its derivatives for  $k = 0, 1, \dots, L$ , and  $i = 0, 1, \dots, d_k$ . Prove that an approximating polynomial of degree  $K = (\sum_{k=0}^L (d_k + 1)) - 1$  can be uniquely determined. Find an error bound for  $\epsilon(t) = x(t) - p_K(t)$ .

5.8. *Proof of Weierstrass Approximation Theorem*

Let  $x(t)$  be continuous on  $[0, 1]$  and let  $\epsilon > 0$ . For each  $K \in \mathbb{Z}^+$ , define the *Bernstein polynomial* of degree  $K$  as

$$p_K(t) = \sum_{k=0}^K \binom{K}{k} x\left(\frac{k}{K}\right) t^k (1-t)^{K-k}.$$

Show that

$$\lim_{K \rightarrow \infty} \|x - p_K\|_\infty = 0.$$

Followed by an appropriate change of variables to the general interval  $[a, b]$ , show this proves Theorem 5.3 [6].

(Hint: TBD.)

5.9. *Near Minimax Approximation*

Prove that

$$\max_{t \in [-1, 1]} \prod_{k=0}^K |t - t_k|$$

is minimized by choosing  $\{t_k\}_{k=0}^K$  to be the  $K + 1$  zeros of the Chebyshev polynomial  $T_{K+1}(t)$ . Show furthermore that the interpolation error bound (5.7b) becomes

$$|x(t) - p_{\text{interp}}(t)| \leq \frac{1}{(K+1)!2^K} \max_{\xi \in [-1, 1]} |x^{(K+1)}(\xi)|$$

for approximation of  $x(t)$  on  $[-1, 1]$  with a polynomial of degree at most  $K$ .

(Hint: This can be deduced from the minimax approximation of  $t^{K+1}$  by  $t^{K+1} - 2^{-K} T_{K+1}(t)$  with approximation error  $2^{-K}$ .)

5.10. *Truncation as Orthogonal Projection*

Using the projection theorem, Theorem 1.26, show that truncation of the ideal filter (5.26) is the least-squares approximation solution.

5.11. *Spline Spaces*

Given is the  $N$ th order B-spline as in (5.32)

- (i) Show that  $\beta^{(N)}(t)$  has  $N - 1$  continuous derivatives, as well as an  $N$ th derivative everywhere except at integers.
- (ii) Show that functions in  $S_N$  belong to  $C^{N-1}$ , as well as to  $C^N$  except at integers.

5.12. *Dual Spline Bases*

Consider (5.31b) with  $N = 2$ , the quadratic spline  $\beta^{(2)}(t)$ . Denote by  $a_n^{(2)}$  its deterministic autocorrelation evaluated at integers.

- (i) Show that  $A^{(2)}(e^{j\omega}) > 0$ .
- (ii) Find the spectral root of  $A^{(2)}(z)$ .
- (iii) Find the inverse  $C(z) = 1/A^{(2)}(z)$  and its stable, two-sided inverse  $z$ -transform  $c_n$  such that

$$\langle c, a^{(2)} \rangle_n = \delta_n.$$

- (iv) Verify that

$$\tilde{\beta}^{(2)}(t) = \sum_{k \in \mathbb{Z}} c_k \beta^{(2)}(t - k)$$

satisfies the biorthogonality relation

$$\langle \tilde{\beta}^{(2)}(t), \beta^{(2)}(t - n) \rangle_t = \delta_n.$$

5.13. *Battle-Lemarié Wavelets*

Consider (5.31b) with  $N = 2$ , the quadratic spline  $\beta^{(2)}(t)$ .

- (i) Form the following  $2\pi$ -periodic function:

$$N(e^{j\omega}) = \sum_{k \in \mathbb{Z}} |B^{(2)}(\omega + 2k\pi)|^2. \quad (\text{P5.13-1})$$

- (ii) Form the following function:

$$\Phi(\omega) = \frac{B^{(2)}(\omega)}{N(e^{j\omega})}. \quad (\text{P5.13-2})$$

Prove that

$$|\Phi(\omega + 2k\pi)|^2 = 1. \quad (\text{P5.13-3})$$

- (iii) Find the inverse Fourier transform of  $\Phi(\omega)$  and prove that

$$\langle \varphi(t), \varphi(t - n) \rangle_t = \delta_n. \quad (\text{P5.13-4})$$

The resulting function,  $\varphi(t)$  is called a *Battle-Lemarié scaling function*.<sup>97</sup> What we have above is a general procedure for orthogonalizing splines, leading to a function  $\varphi(t)$  that, with its integer shifts, form an orthonormal basis for  $S_2$ .

5.14. *Computing Inner Products with Splines*

Consider a function  $x(t)$  that is zero outside of  $[0, L]$ . We want to compute

$$y_n = \langle x(t), \beta^{(N)}(t - n) \rangle_t,$$

where  $\beta^{(N)}(t)$  is the causal  $N$ th order B-spline.

- (i) Show that

$$y_n = \begin{cases} \langle x(t), \beta^{(0)}(t - n) \rangle_t, & 0 \leq n \leq L - 1; \\ 0, & \text{otherwise,} \end{cases} = \begin{cases} X_{n+1} - X_n, & 0 \leq n \leq L - 1; \\ 0, & \text{otherwise,} \end{cases}$$

where

$$X_n = \int_0^n x(t) dt$$

is the primitive of  $x(t)$  evaluated at integers.

- (ii) Generalize the above result to the inner product with an  $N$ th-order spline.

<sup>97</sup>We defer the discussion on wavelets for Chapter 12.

5.15. *Polynomial Reproduction by B-Splines and Their Shifts*

For the  $N$ th-order B-spline,  $\beta^{(N)}(t)$ , prove (5.65).

(Hint: You can use the fact that

$$\sum_n \beta^{(N)}(t - n) = 1.)$$

(i) Take  $k$  derivatives of (5.65) using (5.55) to show that the result is a constant, upper bounding the degree of the polynomial.

(ii) Take  $k$  integrals of (5.65) using (5.59) to show that the result is a polynomial of degree  $k$ .

5.16. *Linear Approximation in a Biorthogonal Basis*

Extend (5.73)–(5.74c) to the case of truncated biorthogonal bases.

5.17. *Linear versus Nonlinear Approximation*

Consider two uncorrelated jointly Gaussian random variables  $x_1 \sim \mathcal{N}(0, 1)$  and  $x_2 \sim \mathcal{N}(0, 1)$ . Similarly to Figure 5.25, compare linear versus nonlinear approximation, and the resulting expected quadratic error. Without any loss of generality, you may use the standard basis  $\varphi_0 = [1 \ 0]^T$ ,  $\varphi_1 = [0 \ 1]^T$ .

5.18. *Quantizer Performance*

Show that (5.96) holds, with a different function  $g$ , for any family of quantizers that can be described by its operation on a normalized variable, not just optimal quantizers.



## Chapter 6

# Time-Frequency Localization

### Contents

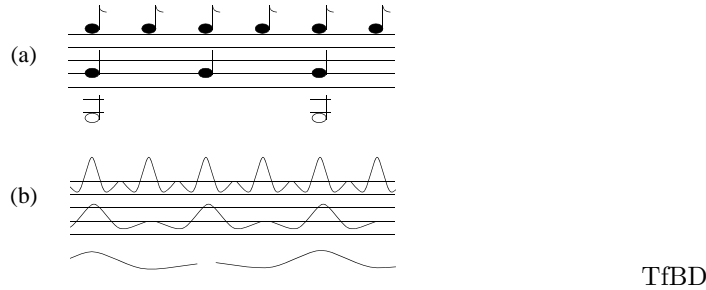
<b>6.1</b>	<b>Introduction . . . . .</b>	<b>526</b>
<b>6.2</b>	<b>Localization for Functions . . . . .</b>	<b>528</b>
<b>6.3</b>	<b>Localization for Sequences . . . . .</b>	<b>536</b>
	<b>Chapter at a Glance . . . . .</b>	<b>545</b>
	<b>Historical Remarks . . . . .</b>	<b>546</b>
	<b>Further Reading . . . . .</b>	<b>546</b>
	<b>Exercises with Solutions . . . . .</b>	<b>546</b>
	<b>Exercises . . . . .</b>	<b>550</b>

In Part II, we will construct various sets of vectors  $\{\varphi_k\}$  to use for signal analysis and synthesis. For a representation

$$x = \sum_k \alpha_k \varphi_k$$

to exist for any  $x$ , we need  $\{\varphi_k\}$  to be complete. For the representation to be unique,  $\{\varphi_k\}$  must be a basis, and we will often construct  $\{\varphi_k\}$  to also be an orthonormal set. However, these properties are not enough for the set  $\{\varphi_k\}$  to be useful. For most applications, the utility of a representation is tied to time, frequency, scale, and resolution properties of the  $\varphi_k$ s. Computational efficiency is also a concern; we begin to address it starting from Chapter 7.

Our primary goal in this chapter is to explore time, frequency, scale, and resolution properties of *individual* basis vectors, as these will be our tools for extracting information about a given signal (function or sequence). We call the process of extracting information *probing*, and we perform it by computing an inner product of the signal with a probe  $\varphi$ . The result (measurement) of probing is the coefficient



(a) Musical score. (b) Time-domain functions. (c) Time-frequency plane for (b).

**Figure 6.1:** Musical score as an illustration of a time-frequency plane [167].

$\alpha$ ; for functions,

$$\begin{aligned}\alpha &= \langle x, \varphi \rangle_t = \int_{-\infty}^{\infty} x(t) \varphi^*(t) dt \\ &\stackrel{(a)}{=} \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) \Phi^*(\omega) d\omega = \frac{1}{2\pi} \langle X, \Phi \rangle_{\omega},\end{aligned}\quad (6.1)$$

where (a) follows from the generalized Parseval's equality (3.69b). The two integral expressions for  $\alpha$  are suggestive of probing  $x$  for its characteristics in time and in frequency. Because of the *uncertainty principle*, the probe will have limited simultaneous localization in time and frequency, so what we learn about  $x$  will be limited as well. The principle holds across various ways to measure time and frequency localization with a single probing, and for both discrete and continuous time. Scaling in time or frequency leads to the trade-off between these localizations in the two domains, while shifting and modulation leave them unchanged. Overall, the uncertainty principle in its various guises helps us understand time and frequency representations and localization properties of individual vectors contributing to our intuition for what can be expected of a basis.

## 6.1 Introduction

For certain simple signals, time and frequency properties are quite intuitive. Think of a note on a musical score. It is of a certain frequency (for example, middle A has the frequency of 440 Hz), it has a start time, and its value ( $\flat$ ,  $\natural$ ,  $\sharp$ ) indicates its relative duration. We can think of the musical score as a time-frequency plane with a logarithmic frequency axis and notes as rectangles in that plane with horizontal extent determined by start and end times, and vertical position related in some way to frequency, as in Figure 6.1.

**Localization in Time and Frequency** Time and frequency views of a signal are intertwined in several ways. For example, the Fourier transform gives a precise

sense of interchangeability: if  $x(t)$  has the Fourier transform  $X(\omega)$ , then  $X(t)$  has the Fourier transform  $2\pi x(-\omega)$ . More relevantly to this chapter, various forms of the uncertainty principle determine the trade-off between fine localization in the two domains; signals finely localized in time will be coarsely localized in frequency; conversely, signals finely localized in frequency will be coarsely localized in time. The uncertainty principle also bounds the product of spreads in time and frequency, with the lower bound reached by Gaussian functions.

**Scale** Another natural notion for signals is scale. For example, given a portrait of a person, recognizing that person should not depend on whether they occupy one-tenth or one-half of the image;<sup>98</sup> thus, image recognition should be scale invariant. Signals that are scales of each other are often considered as equivalent. However, this scale invariance is a purely continuous-time property, since discrete-time sequences cannot be rescaled easily. For example, downsampling by a factor of  $N$  is in general a lossy operation, while upsampling by a factor of  $N$  introduces spurious zeros. What we will consider natural scaling operations in discrete domain are sampling (lowpass prefiltering followed by downsampling) and interpolation (upsampling followed by lowpass postfiltering) operations we introduced in Chapter 4.

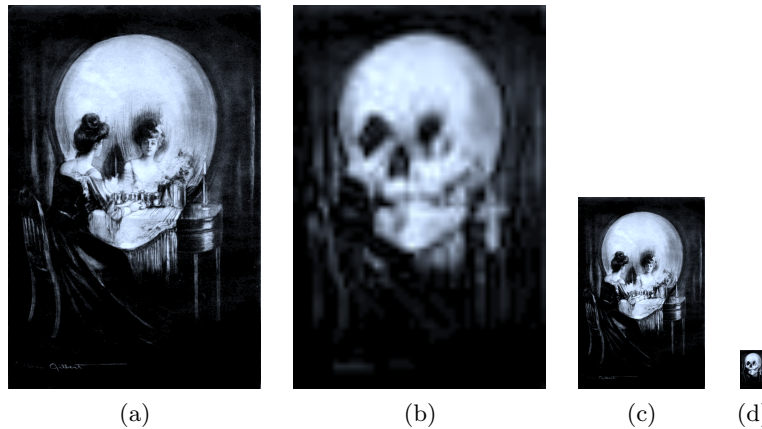
**Resolution** A final important notion we discuss is that of resolution. Intuitively, a blurred photograph does not have the resolution of a sharp one, even when the two prints are of the same physical size. Thus, resolution is related to the bandwidth of a signal, or, more generally, to the number of degrees of freedom per unit time (or space). Classical bandwidth is then proportional to resolution. Consider the space of bandlimited functions  $x(t) \in \text{BL}[-\omega_0/2, \omega_0/2]$  as in Definition 4.12. Then, the sampling theorem, Theorem 4.14, states that samples taken every  $T = 2\pi/\omega_0$  sec, or  $x_n = x(nT)$ ,  $n \in \mathbb{Z}$ , uniquely specify  $x(t)$ . In other words, real functions of bandwidth  $\omega_0$  have  $\omega_0/(2\pi)$  real degrees of freedom per unit time.

An example of a set of functions that, while not bandlimited, do have a finite number of degrees of freedom per unit time, are piecewise-constant functions from (4.1). Clearly,  $x(t)$  has 1 degree of freedom per unit time, but an unbounded spectrum since it is discontinuous at every integer. This function is part of a general class of functions belonging to the so-called shift-invariant subspaces we studied in Chapter 4 (see also Exercise 6.1).

**Interactions** Clearly, scale and resolution interact; this is most obvious with images as illustrated on a drawing by C. Allan Gilbert entitled *All is Vanity*,<sup>99</sup> Figure 6.2. It is designed to be perceived as either a woman sitting in front of a mirror (when seen from near by and at high resolution as in Figure 6.2(a)), or as a skull (when seen at low resolution as in Figure 6.2(b)). Figure 6.2(c) illustrates the notion of a change of scale; even though the scale has been halved, resolution remains

<sup>98</sup>This is true within some bounds, linked to resolution as we discuss shortly.

<sup>99</sup>Another beautiful optical illusion is Salvador Dali's *Gala Contemplating the Mediterranean Sea which at Twenty Meters becomes a Portrait of Abraham*; the title says it all.



**Figure 6.2:** *All is Vanity* by C. Allan Gilbert illustrates notions of scale and resolution for an image. (a) The original, high-resolution, version. (b) A blurred, lower-resolution, version; resolution is lower and thus our perception of the image has changed. (c) A scaled version of half size in each dimension; resolution is unchanged and thus our perception of the image remains unchanged, at least as long as our visual acuity can pick up the necessary resolution. (d) A scaled version, where the visual acuity is not sufficient to see the full information in the image.

unchanged and thus our perception of the image as long as our perception is sufficiently good. Figure 6.2(d) shows a poststamp version, where we cannot see the details anymore, and thus, only the skull is left.

Filtering can affect resolution as well. If a function of bandwidth  $\omega_0$  is perfectly lowpass filtered to  $|\omega| < \beta\omega_0/2$ , with  $0 < \beta < 1$ , then its resolution changes from  $\omega_0/2\pi$  to  $\beta\omega_0/2\pi$ . The same holds for sequences, where an ideal lowpass filter with bandwidth  $\beta\pi$ ,  $0 < \beta < 1$ , reduces the resolution to  $\beta/2$  samples per unit time.

## Chapter Outline

The present chapter explores the above basic notions as well as their interactions in detail. Section 6.2 discusses localization concepts for functions, while Section 6.3 does the same for sequences with a brief mention of finite-length sequences.

## 6.2 Localization for Functions

### 6.2.1 Time Localization

Consider a function  $x(t) \in \mathcal{L}^2(\mathbb{R})$  where  $t$  is a time index. We now discuss localization of the function in time. When the function is finitely supported, its Fourier transform is not (it can only have isolated zeros); that is, a function cannot be perfectly localized in both time and frequency. Even if not of finite support, a function might still decay rapidly as  $t \rightarrow \pm\infty$ . Such decay is necessary for working in  $\mathcal{L}^2(\mathbb{R})$ ;



## 6.2. Localization for Functions

529

the function must decay faster than  $|t|^{-1/2}$  for large  $t$  (see Section 3.4.2).

A concise way to describe locality (or lack thereof), is to introduce a spreading measure akin to standard deviation, requiring normalization so that  $|x(t)|^2/\|x\|^2$  can be interpreted as a PDF (this normalization is precisely the same as restricting attention to unit-norm functions). Its mean is then the time center of the function and its standard deviation is the time spread.

**DEFINITION 6.1 (TIME CENTER AND SPREAD FOR FUNCTIONS)** Let  $x(t)$  be a function in  $\mathcal{L}^2(\mathbb{R})$  of norm  $\|x\|^2$ .

Its time center  $\mu_t$  and time spread  $\Delta_t$  are

$$\mu_t = \frac{1}{\|x\|^2} \int_{-\infty}^{\infty} t |x(t)|^2 dt, \quad (6.2a)$$

$$\Delta_t^2 = \frac{1}{\|x\|^2} \int_{-\infty}^{\infty} (t - \mu_t)^2 |x(t)|^2 dt. \quad (6.2b)$$

**EXAMPLE 6.1 (TIME SPREADS FOR FUNCTIONS)** Consider the following functions and their time spreads (see Figure 3.9):

- (i) The sinc function from (3.75) has  $\mu_t = 0$  and infinite  $\Delta_t^2$ , as  $|x(t)|^2$  decays only as  $|t|^{-2}$ .
- (ii) The box function from (3.76) has  $\mu_t = 0$  and  $\Delta_t^2 = t_0^2/12$ .
- (iii) The Gaussian function from (3.78) with  $\gamma = (2\alpha/\pi)^{1/4}$  is of unit  $\mathcal{L}^2$  norm; see (3.13c). It has  $\mu_t = 0$  and  $\Delta_t^2 = 1/(4\alpha)$ .

From the above example, we see that the time spread can vary widely; the box function has the narrowest one, while the sinc function has an infinite spread, showing that it can be unbounded, even for widely-used functions. Exercise 6.1 explores functions based on the box function and convolutions thereof.

The time center and spread satisfy the following (see Solved Exercise 6.2):

- (i) With the shift of a function in time,  $y(t) = x(t - t_0)$ ,

$$\mu_{y,t} = \mu_{x,t} + t_0, \quad (6.3a)$$

$$\Delta_{y,t} = \Delta_{x,t}, \quad (6.3b)$$

that is, the time center shifts and the time spread is invariant.

- (ii) With the norm-preserving scaling of a function in time,

$$y(t) = \sqrt{\alpha} x(\alpha t), \quad (6.3c)$$

the time center and spread satisfy

$$\mu_{y,t} = \frac{1}{\alpha} \mu_{x,t}, \quad (6.3d)$$

$$\Delta_{y,t} = \frac{1}{\alpha} \Delta_{x,t}, \quad (6.3e)$$

that is, both the time center and the time spread scale.

### 6.2.2 Frequency Localization

The concept dual to time localization of  $x(t)$  is frequency localization of its Fourier transform  $X(\omega)$ . Similarly to time, when  $X(\omega)$  is finitely supported (bandlimited as in Definition 4.12), the function in time is not. If not of finite support, the Fourier transform might still decay rapidly, leading to the notion of the frequency spread. As we did for the time spread, we normalize the frequency spread, so as to be able to interpret  $|X(\omega)|^2/\|X\|^2$  as a PDF.

**DEFINITION 6.2 (FREQUENCY CENTER AND SPREAD FOR FUNCTIONS)** Let  $x(t)$  be a function in  $\mathcal{L}^2(\mathbb{R})$  with the Fourier transform  $X(\omega)$  of norm  $\|X\|^2 = 2\pi\|x\|^2$ .

Its frequency center  $\mu_\omega$  and frequency spread  $\Delta_\omega$  are

$$\mu_\omega = \frac{1}{2\pi\|x\|^2} \int_{-\infty}^{\infty} \omega |X(\omega)|^2 d\omega, \quad (6.4a)$$

$$\Delta_\omega^2 = \frac{1}{2\pi\|x\|^2} \int_{-\infty}^{\infty} (\omega - \mu_\omega)^2 |X(\omega)|^2 d\omega. \quad (6.4b)$$

Note that the frequency center will be 0 for all real functions because of the symmetry of the Fourier transform.

**EXAMPLE 6.2 (FREQUENCY SPREADS FOR FUNCTIONS)** We consider the same functions as in Example 6.1 and their frequency spreads (see Figure 3.9):

- (i) The Fourier transform of the sinc function from (3.75) has  $\mu_\omega = 0$  and  $\Delta_\omega^2 = \omega_0^2/12$ .
- (ii) The Fourier transform of the box function from (3.76) has  $\mu_\omega = 0$  and infinite  $\Delta_\omega^2$ , as  $|X(\omega)|^2$  decays only as  $|\omega|^{-2}$ .
- (iii) The Fourier transform of the Gaussian function from (3.78) has  $\mu_\omega = 0$  and  $\Delta_\omega^2 = \alpha$ .

The frequency center and spread satisfy the following (see Solved Exercise 6.2):

- (i) With the shift of a function in frequency,  $Y(\omega) = X(\omega - \omega_0)$ ,

$$\mu_{y,\omega} = \mu_{x,\omega} + \omega_0, \quad (6.5a)$$

$$\Delta_{y,\omega} = \Delta_{x,\omega}, \quad (6.5b)$$

that is, the frequency center shifts and the frequency spread is invariant.

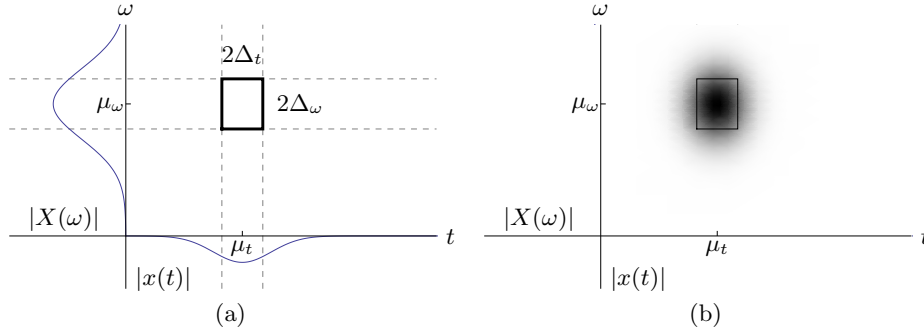
- (ii) With the scaling of a function in frequency,  $Y(\omega) = (1/\sqrt{\alpha}) X(\omega/\alpha)$ ,<sup>100</sup>

$$\mu_{y,\omega} = \alpha \mu_{x,\omega}, \quad (6.5c)$$

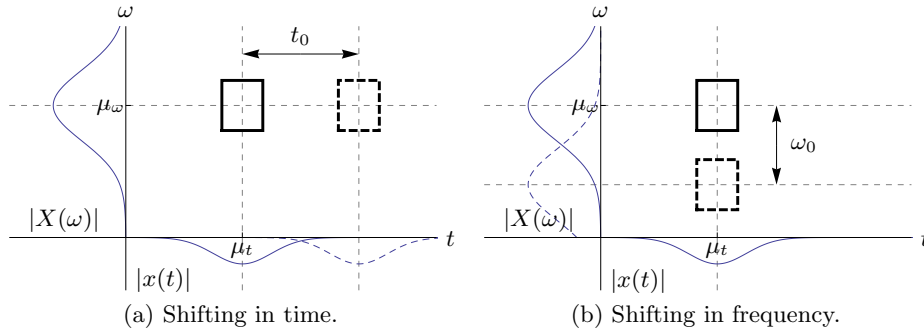
$$\Delta_{y,\omega} = \alpha \Delta_{x,\omega}, \quad (6.5d)$$

that is, both the frequency center and the frequency spread scale.

<sup>100</sup>We choose this scaling to be consistent with the scaling in time,  $y(t) = \sqrt{\alpha} x(\alpha t)$ , since then its Fourier transform is scaled in frequency,  $Y(\omega) = (1/\sqrt{\alpha}) X(\omega/\alpha)$ , as in (3.58b).



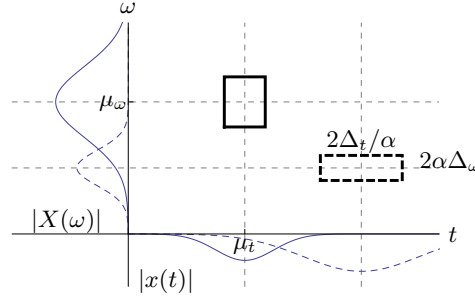
**Figure 6.3:** The time-frequency plane. (a) The function  $x(t)$  with the Fourier transform  $X(\omega)$  has an associated Heisenberg box centered at  $(\mu_t, \mu_\omega)$  of width  $2\Delta_t$  and height  $2\Delta_\omega$ . (b) The product  $|x(t)|^2 |X(\omega)|^2$  as a density plot.



**Figure 6.4:** Shifting of a function with the Heisenberg box centered at  $(\mu_t, \mu_\omega)$  of width  $2\Delta_t$  and height  $2\Delta_\omega$ . (a) Shifting in time by  $t_0$ ,  $x(t - t_0)$ , shifts the Heisenberg box to  $(\mu_t + t_0, \mu_\omega)$ ; the size remains the same. (b) Shifting in frequency by  $\omega_0$  (modulating),  $X(\omega - \omega_0)$ , shifts the Heisenberg box to  $(\mu_t, \mu_\omega + \omega_0)$ ; the size remains the same.

### 6.2.3 Uncertainty Principle for Functions

**Heisenberg Box** Given a function  $x(t)$  and its Fourier transform  $X(\omega)$ , we have just introduced the 4-tuple  $(\mu_t, \Delta_t, \mu_\omega, \Delta_\omega)$ , describing the function's center in time and frequency  $(\mu_t, \mu_\omega)$  and its spread in time and frequency  $(\Delta_t, \Delta_\omega)$ . It is convenient to show this pictorially (see Figure 6.3), as it conveys the idea that there is a center of mass  $(\mu_t, \mu_\omega)$  around which a rectangular box of width  $2\Delta_t$  and height  $2\Delta_\omega$  is located. The plane on which this is drawn is called the *time-frequency plane*, and the box is usually called a *Heisenberg box*, or a *time-frequency tile*. We adopt a convention that we show only the first quadrant of the time-frequency plane; we reserve the second quadrant for the magnitude response of the Fourier transform of the function,  $|X(\omega)|$ , and the fourth quadrant for the absolute value of the function itself,  $|x(t)|$ .



**Figure 6.5:** Scaling of a function with the Heisenberg box centered at  $(\mu_t, \mu_\omega)$  of width  $2\Delta_t$  and height  $2\Delta_\omega$ , shifts the Heisenberg box to  $(\mu_t/\alpha, \alpha\mu_\omega)$  and scales its width to  $2\Delta_t/\alpha$  and its height to  $2\alpha\Delta_\omega$ .

From our previous discussion on time and frequency shifting, we know that a function obtained by a shift and modulation of  $x(t)$ ,

$$e^{j\omega_0 t} x(t - t_0) \xleftrightarrow{\text{FT}} e^{-j\omega t_0} X(\omega - \omega_0), \quad (6.6)$$

will have a Heisenberg box of the same size, simply shifted by  $t_0$  and  $\omega_0$  (dashed boxes in Figure 6.4; we separated the effects of time and frequency shifts for clarity).

Similarly, a function obtained by scaling of  $x(t)$ ,

$$\sqrt{\alpha} x(\alpha t) \xleftrightarrow{\text{FT}} \frac{1}{\sqrt{\alpha}} X\left(\frac{\omega}{\alpha}\right), \quad (6.7)$$

will have a Heisenberg box of the same area, scaled appropriately in time and frequency. That is, if  $x(t)$  has a Heisenberg box specified by  $(\mu_t, \Delta_t, \mu_\omega, \Delta_\omega)$ , then the scaled function has a Heisenberg box specified by  $(\mu_t/\alpha, \Delta_t/\alpha, \alpha\mu_\omega, \alpha\Delta_\omega)$  (dashed box in Figure 6.5). The effects of shift, modulation and scaling on the Heisenberg boxes are summarized in Table 6.1.

Function	Time center	Time spread	Fourier transf.	Freq. center	Freq. spread
$x(t)$	$\mu_t$	$\Delta_t$	$X(\omega)$	$\mu_\omega$	$\Delta_\omega$
$x(t - t_0)$	$\mu_t + t_0$	$\Delta_t$	$e^{-j\omega t_0} X(\omega)$	$\mu_\omega$	$\Delta_\omega$
$e^{j\omega_0 t} x(t)$	$\mu_t$	$\Delta_t$	$X(\omega - \omega_0)$	$\mu_\omega + \omega_0$	$\Delta_\omega$
$\sqrt{\alpha} x(\alpha t)$	$\mu_t/\alpha$	$\Delta_t/\alpha$	$X(\omega/\alpha)/\sqrt{\alpha}$	$\alpha\mu_\omega$	$\alpha\Delta_\omega$

**Table 6.1:** Effect of a shift in time and frequency, as well as scaling in time and frequency, on a Heisenberg box  $(\mu_t, \Delta_t, \mu_\omega, \Delta_\omega)$ .

**Uncertainty Principle** So far, we have considered the effect of shifting in time and frequency as well as scaling on a Heisenberg box. What about the effect on the area

## 6.2. Localization for Functions

533

of the Heisenberg box? The intuition, corroborated by what we saw with scaling, is that one can trade time spread for frequency spread. Moreover, from Examples 6.1 and 6.2, we know that a function that is narrow in one domain will be broad in the other. It is thus intuitive that the size of the Heisenberg box is lower bounded, so that no function can be arbitrarily narrow in both time and frequency.

**THEOREM 6.3 (UNCERTAINTY PRINCIPLE)** Let  $x \in \mathcal{L}^2(\mathbb{R})$ , with the Fourier transform  $X(\omega)$ , the time spread  $\Delta_t$  and frequency spread  $\Delta_\omega$ . Then,

$$\Delta_t^2 \Delta_\omega^2 \geq \frac{1}{4}, \quad (6.8)$$

with the lower bound attained by Gaussian functions (3.78).

*Proof.* We prove the theorem for real functions; see Exercise 6.2 for the complex case. Without loss of generality, assume that  $x(t)$  is centered at  $t = 0$  and is of unit norm,  $\|x\|^2 = 1$ ; otherwise, we may shift and scale it appropriately. Since  $x(t)$  is real,  $X(\omega)$  is centered at  $\omega = 0$ , so  $\mu_t = \mu_\omega = 0$ .

Suppose  $x(t)$  has a bounded derivative  $x'(t)$ ; if not,  $\Delta_\omega^2 = \infty$  so the statement holds trivially. Consider the function  $tx(t)x'(t)$  and its integral. Using the Cauchy–Schwarz inequality (1.31), we can write

$$\begin{aligned} \left| \int_{-\infty}^{\infty} tx(t)x'(t) dt \right|^2 &\leq \int_{-\infty}^{\infty} |tx(t)|^2 dt \int_{-\infty}^{\infty} |x'(t)|^2 dt \\ &\stackrel{(a)}{=} \underbrace{\int_{-\infty}^{\infty} |tx(t)|^2 dt}_{\Delta_t^2} \underbrace{\frac{1}{2\pi} \int_{-\infty}^{\infty} |j\omega X(\omega)|^2 d\omega}_{\Delta_\omega^2} = \Delta_t^2 \Delta_\omega^2, \end{aligned} \quad (6.9)$$

where (a) follows from Parseval's equality (3.69a) and the differentiation in frequency property of the Fourier transform (3.61a). We now simplify the left side:

$$\int_{-\infty}^{\infty} tx(t)x'(t) dt \stackrel{(a)}{=} \frac{1}{2} \int_{-\infty}^{\infty} t \frac{dx^2(t)}{dt} dt \stackrel{(b)}{=} \frac{1}{2} t x^2(t) \Big|_{-\infty}^{\infty} - \frac{1}{2} \underbrace{\int_{-\infty}^{\infty} x^2(t) dt}_1 \stackrel{(c)}{=} -\frac{1}{2},$$

where (a) follows from  $(x^2(t))' = 2x'(t)x(t)$ ; (b) from integration by parts; and (c) holds because  $x(t) \in \mathcal{L}^2(\mathbb{R})$  implies that it decays faster than  $1/\sqrt{|t|}$  for  $t \rightarrow \pm\infty$ , and thus  $\lim_{t \rightarrow \pm\infty} tx^2(t) = 0$  (see Section 3.4.2). Substituting this into (6.9) yields (6.8).

To find functions that meet the bound with equality, recall that Cauchy–Schwarz inequality becomes an equality if and only if the two functions are collinear (scalar multiples of each other), or at least one of them is 0. In our case this means  $x'(t) = \beta tx(t)$ . Functions satisfying this relation have the form  $x(t) = \gamma e^{\beta t^2/2} = \gamma e^{-\alpha t^2}$ , the Gaussian functions.

The uncertainty principle points to a fundamental limitation in time-frequency analysis using linear analysis using inner products.<sup>101</sup> If we desire to analyze a

<sup>101</sup>There exist nonlinear techniques that are not bound by this limiting factor; we discuss these in Chapter 13.

TfBD    TfBD  
(a)    (b)

**Figure 6.6:** Chirp signal and time-frequency analysis with a linearly increasing frequency. (a) An example of a windowed chirp with a linearly increasing frequency. Real and imaginary parts are shown as well as the raised cosine window. (b) Idealized time-frequency analysis.

TfBD    TfBD  
(a)    (b)

**Figure 6.7:** Analysis of a chirp signal. (a) One of the analyzing functions, consisting of a (complex) modulated window. (b) Magnitude squared of the inner products between the chirp and various shifts and modulates of the analyzing functions, showing a blurred version of the chirp.

function with a *probing function* to extract information about the function around a location  $(\mu_t, \mu_\omega)$ , the probing function will necessarily be imprecise. That is, if we wish very precise frequency information about the function, its time location will be uncertain, and vice versa.

**EXAMPLE 6.3 (CHIRP FUNCTION)** As an example of time-frequency analysis and the trade-off between time and frequency sharpness, consider a windowed chirp function (complex exponential with a rising frequency). Instead of a fixed frequency  $\omega_0$ , the *local frequency* linearly grows with time,  $\omega_0 t$ ,

$$x(t) = w(t) e^{j\omega_0 t^2},$$

where  $w(t)$  is an appropriate window function.<sup>102</sup> Figure 6.6(a) shows an example of a chirp function with an idealized time-frequency analysis in Figure 6.6(b).

As analyzing functions, we choose windowed complex exponentials (but with a fixed frequency). The choice we have is the size of the window (assuming the analyzing function will cover all shifts and modulations of interest). A long window allows for sharp frequency analysis, however, no frequency is really present other than at one instant. A short window will do justice to the transient nature of *frequency* in the chirp, but will only give a very approximate frequency analysis due to the uncertainty principle. A compromise between time and frequency sharpness must be sought, and one such possible analysis is shown in Figure 6.7.

**Other Localization Measures** While the uncertainty principle uses a spreading measure akin to standard deviation, other measures can be defined. Though they typically lack fundamental bounds of the kind given by the uncertainty principle

<sup>102</sup>Bats use such chirp functions to hunt for bugs.

(6.8), they can be quite useful as well as intuitive. One such measure, easily applicable to functions that are symmetric in both time and frequency, finds the centered intervals containing a given percentage  $\beta$  of the energy in time and frequency (where  $\beta$  is typically 0.90 or 0.95).

For a unit-norm function  $x(t)$  symmetric around  $\mu_t$ , with the  $2\pi$ -norm Fourier transform  $X(\omega)$  symmetric around  $\mu_\omega$ , the time spread  $\hat{\Delta}_t^{(\beta)}$  and frequency spread  $\hat{\Delta}_\omega^{(\beta)}$  are now defined such that

$$\int_{\mu_t - \frac{1}{2}\hat{\Delta}_t^{(\beta)}}^{\mu_t + \frac{1}{2}\hat{\Delta}_t^{(\beta)}} |x(t)|^2 dt = \beta, \quad (6.10a)$$

$$\frac{1}{2\pi} \int_{\mu_\omega - \frac{1}{2}\hat{\Delta}_\omega^{(\beta)}}^{\mu_\omega + \frac{1}{2}\hat{\Delta}_\omega^{(\beta)}} |X(\omega)|^2 d\omega = \beta. \quad (6.10b)$$

Exercise 6.3 shows that  $\hat{\Delta}_t^{(\beta)}$  and  $\hat{\Delta}_\omega^{(\beta)}$  satisfy the same shift, modulation and scaling behavior as do  $\Delta_t$  and  $\Delta_\omega$  in Table 6.1.

### 6.2.4 Scale Localization

At the very beginning of the chapter, we discussed the idea of signal analysis as computing an inner product with a probing function. Along with the Heisenberg box 4-tuple, another key property of a probing function is its scale. Scale is closely related to the time spread, but it is inherently a relative (rather than absolute) quantity. Before further describing scale, let us revisit scaling; we will point out the fundamental difference between continuous- and discrete-time scaling operations in the next section.

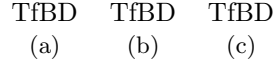
An energy-conserving rescaling of  $x(t)$  by a factor  $\alpha \in \mathbb{R}^+$  was given in (6.7). Clearly, this is a reversible process, since rescaling  $y(t)$  by  $(1/\alpha)$  gives

$$\frac{1}{\sqrt{\alpha}} y\left(\frac{t}{\alpha}\right) = x(t).$$

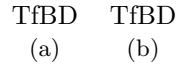
We can gain an intuitive understanding of scale by examining maps. The usual notion of scale in maps is the following: in a map at scale 1:100,000, an object of length 1 km is represented by a length of  $(10^3\text{m})/10^5 = 1$  cm. That is, the scale factor  $\alpha = 10^5$  is used as a contraction factor, to map a reality  $x(t)$  into a scaled version  $y(t) = \sqrt{\alpha}x(\alpha t)$  (with the energy normalization factor  $\sqrt{\alpha}$  of no real significance because reality and a map are not of the same dimension). However, reality does provide us with something important: a baseline scale against which to compare the map.

When we look at functions in  $\mathcal{L}^2(\mathbb{R})$ , a baseline scale does not necessarily exist. When  $y(t) = \sqrt{\alpha}x(\alpha t)$ , we say that  $y$  is at a larger scale if  $\alpha > 1$ , and at a smaller scale if  $\alpha \in (0, 1)$ . There is no absolute scale for  $y$  unless we arbitrarily define a scale for  $x$ .

Now consider the use of a real probing function  $\varphi(t)$  to extract some information about  $x(t)$ . If we compute the inner product between the probing function and



**Figure 6.8:** Aerial photographs of the EPF Lausanne campus at various scales. (a) 5,000. (b) 10,000. (c) 20,000.



**Figure 6.9:** Signals with features at different scales require probing functions adapted to the scales of those features. (a) A wide-area feature requires a wide probing function. (b) A narrow-area feature requires a sharp probing function.

a scaled function, we get

$$\begin{aligned} \langle \sqrt{\alpha} x(\alpha t), \varphi(t) \rangle &= \sqrt{\alpha} \int_{-\infty}^{\infty} x(\alpha t) \varphi(t) dt = \frac{1}{\sqrt{\alpha}} \int_{-\infty}^{\infty} x(\tau) \varphi\left(\frac{\tau}{\alpha}\right) d\tau \\ &= \langle x(t), \frac{1}{\sqrt{\alpha}} \varphi(t/\alpha) \rangle. \end{aligned} \quad (6.11)$$

Probing a contracted function is equivalent to stretching the probe, thus emphasizing that scale is relative. If only stretched and contracted versions of a single probe are available, large-scale features in  $x(t)$  are seen using stretched probing functions, while small-scale features (fine details) in  $x(t)$  are seen using contracted probing functions.

In summary, large scales  $\alpha \gg 1$  correspond to contracted versions of reality, or to widely-spread probing functions. This duality is inherent in the inner product (6.11). Figure 6.8 shows an aerial photograph with different scale factors as per our convention, while Figure 6.9 shows the interaction of signals with different-size features and probing functions.

### 6.3 Localization for Sequences

Thus far, we have restricted our attention to the study of localization properties and bounds for functions. Analogous results for sequences are not as elegant nor parallel, except in the case of strictly lowpass sequences. Thus, the uncertainty principle we present here holds only for those sequences in  $\ell^2(\mathbb{Z})$  whose DTFT is strictly lowpass in nature, that is, when  $X(e^{j\pi}) = 0$ .

#### 6.3.1 Time Localization

Consider a sequence  $x_n \in \ell^2(\mathbb{Z})$  where  $n$  is a discrete-time index. We now discuss localization of the sequence in time. Similarly to functions, when the sequence is finitely supported, its DTFT is not (it can only have isolated zeros); that is, a



## 6.3. Localization for Sequences

537

sequence cannot be perfectly localized in both time and frequency (see Exercise 6.4). Even if not of finite support, a sequence might still decay rapidly as  $n \rightarrow \pm\infty$ . Similarly to functions, we describe locality concisely by introducing the time center and the time spread.

**DEFINITION 6.4 (TIME CENTER AND SPREAD FOR SEQUENCES)** Let  $x_n$  be a sequence in  $\ell^2(\mathbb{Z})$  of norm  $\|x\|^2$ .

Its time center  $\mu_n$  and time spread  $\Delta_n$  are

$$\mu_n = \frac{1}{\|x\|^2} \sum_{n \in \mathbb{Z}} n |x_n|^2, \quad (6.12a)$$

$$\Delta_n^2 = \frac{1}{\|x\|^2} \sum_{n \in \mathbb{Z}} (n - \mu_n)^2 |x_n|^2. \quad (6.12b)$$

**EXAMPLE 6.4 (TIME SPREADS FOR SEQUENCES)** Consider the following sequences and their time spreads:

- (i) The sinc sequence from Table 3.6 has  $\mu_n = 0$  and infinite  $\Delta_n^2$ , as  $|x_n|^2$  decays only as  $|n|^{-2}$ .
- (ii) The box sequence from Table 3.6 has  $\mu_n = 0$  and  $\Delta_n^2 = (n_0^2 - 1)/12$ .

As for functions, the example shows how time spreads can vary widely.

The time center and spread satisfy the following (see Exercise 6.5):

- (i) With the shift of a sequence in time,  $y_n = x_{n-n_0}$ ,

$$\mu_{y,n} = \mu_{x,n} + n_0, \quad (6.13a)$$

$$\Delta_{y,n} = \Delta_{x,n}, \quad (6.13b)$$

that is, the time center shifts and the time spread is invariant.

- (ii) With the upsampling of a sequence in time followed by lowpass postfiltering,  $y_n = (1/\sqrt{N}) \text{sinc}(\pi n/N) * x_{n/N}$ ,

$$\mu_{y,n} = N \mu_{x,n}, \quad (6.13c)$$

$$\Delta_{y,n} = N \Delta_{x,n}, \quad (6.13d)$$

that is, both the time center and the time spread scale.

With the downsampling of a bandlimited sequence,  $x \in \text{BL}[-\pi/N, \pi/N]$ ,  $y_n = x_{Nn}$ ,

$$\mu_{y,n} = \frac{1}{N} \mu_{x,n}, \quad (6.13e)$$

$$\Delta_{y,n} = \frac{1}{N} \Delta_{x,n}, \quad (6.13f)$$

that is, both the time center and the time spread scale.

If the sequence is not bandlimited, downsampling by  $N$  will keep only its 0th polyphase component. Since its norm is sequence dependent, we cannot tell how the downsampled sequence will be scaled.

### 6.3.2 Frequency Localization

**DEFINITION 6.5 (FREQUENCY CENTER AND SPREAD FOR SEQUENCES)** Let  $x_n$  be a sequence in  $\ell^2(\mathbb{Z})$  with the DTFT  $X(e^{j\omega})$  of norm  $\|X\|^2 = 2\pi\|x\|^2$ .

Its frequency center  $\mu_\omega$  and frequency spread  $\Delta_\omega$  are

$$\mu_\omega = \frac{1}{2\pi\|x\|^2} \int_{-\pi}^{\pi} \omega |X(e^{j\omega})|^2 d\omega, \quad (6.14a)$$

$$\Delta_\omega^2 = \frac{1}{2\pi\|x\|^2} \int_{-\pi}^{\pi} (\omega - \mu_\omega)^2 |X(e^{j\omega})|^2 d\omega. \quad (6.14b)$$

Note that the frequency center will be 0 for all real sequences because of the symmetry of the DTFT.

**EXAMPLE 6.5 (FREQUENCY SPREADS FOR SEQUENCES)** We consider the same sequences as in Example 6.4 and their frequency spreads:

- (i) The DTFT of the sinc sequence from Table 3.6 has  $\mu_\omega = 0$  and  $\Delta_\omega^2 = \omega_0^2/12$ .
- (ii) The DTFT of the box sequence from Table 3.6 has  $\mu_\omega = 0$ ; its  $\Delta_\omega^2$  is no longer infinite, as it is the Dirichlet kernel (see Figure 4.36).

The frequency center and spread satisfy the following (see Exercise 6.5):

- (i) With the shift of the function in frequency,  $Y(e^{j\omega}) = X(e^{j(\omega-\omega_0)})$ ,

$$\mu_{y,\omega} = \mu_{x,\omega} + \omega_0, \quad (6.15a)$$

$$\Delta_{y,\omega} = \Delta_{x,\omega}, \quad (6.15b)$$

that is, the frequency center shifts and the frequency spread is invariant.

- (ii) With the upsampling of the sequence in time followed by lowpass postfiltering,  $Y(e^{j\omega}) = \sqrt{N}X(e^{jN\omega})$  for  $|\omega| \leq \pi/N$ ,

$$\mu_{y,\omega} = \frac{1}{N} \mu_{x,\omega}, \quad (6.15c)$$

$$\Delta_{y,\omega} = \frac{1}{N} \Delta_{x,\omega}, \quad (6.15d)$$

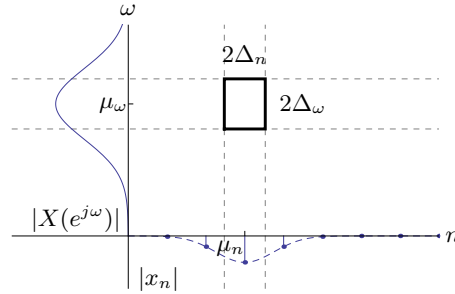
that is, both the frequency center and the frequency spread scale.

With the downsampling of a bandlimited sequence,  $x \in \text{BL}[-\pi/N, \pi/N]$ ,

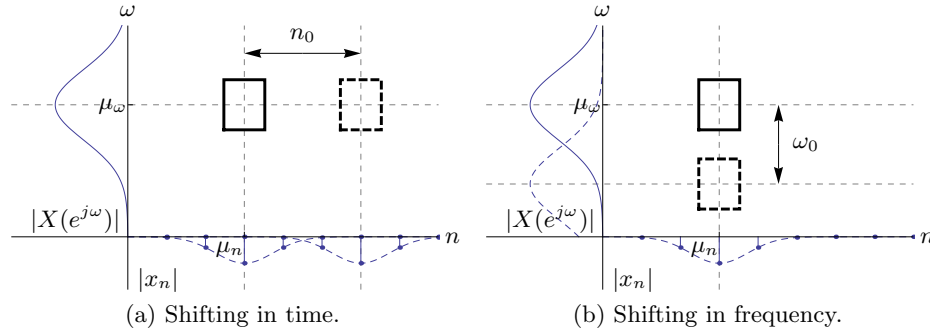
$$\mu_{y,\omega} = N \mu_{x,\omega}, \quad (6.15e)$$

$$\Delta_{y,\omega} = N \Delta_{x,\omega}, \quad (6.15f)$$

that is, both the frequency center and the frequency spread scale.



**Figure 6.10:** The time-frequency plane. The sequence  $x_n$  with the DTFT  $X(e^{j\omega})$  has an associated Heisenberg box centered at  $(\mu_n, \mu_\omega)$  of width  $2\Delta_n$  and height  $2\Delta_\omega$ .



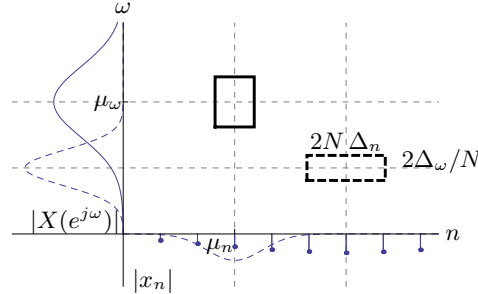
**Figure 6.11:** Shifting of a sequence with the Heisenberg box centered at  $(\mu_n, \mu_\omega)$  of width  $2\Delta_n$  and height  $2\Delta_\omega$ . (a) Shifting in time by  $n_0$ ,  $x_{n-n_0}$ , shifts the Heisenberg box to  $(\mu_n + n_0, \mu_\omega)$ ; the size remains the same. (b) Shifting in frequency by  $\omega_0$  (modulating),  $X(e^{j(\omega-\omega_0)})$ , shifts the Heisenberg box to  $(\mu_n, \mu_\omega + \omega_0)$ ; the size remains the same.

### 6.3.3 Uncertainty Principle for Sequences

**Heisenberg Box** Similarly to functions, given a sequence  $x_n$  and its DTFT  $X(e^{j\omega})$ , we have just introduced the 4-tuple  $(\mu_n, \Delta_n, \mu_\omega, \Delta_\omega)$ , describing the sequence's center in time and frequency  $(\mu_n, \mu_\omega)$  and its spread in time and frequency  $(\Delta_n, \Delta_\omega)$ . As before, we show this pictorially (see Figure 6.10), with the center of mass  $(\mu_n, \mu_\omega)$  around which a rectangular box of width  $2\Delta_n$  and height  $2\Delta_\omega$  is located, again producing a Heisenberg box, but this time for sequences. As before, we show only the first quadrant of the time-frequency plane; we reserve the second quadrant for the magnitude response of the DTFT of the function,  $|X(e^{j\omega})|$ , and the fourth quadrant for the absolute value of the function itself,  $|x(t)|$ .

From our previous discussion on time and frequency shifting, we know that a sequence obtained by a shift and modulation of  $x_n$ ,

$$e^{j\omega_0 n} x_{n-n_0} \xrightarrow{\text{DTFT}} e^{-j\omega t n_0} X(e^{j(\omega-\omega_0)}), \quad (6.16)$$



**Figure 6.12:** Upsampling of a sequence with the Heisenberg box centered at  $(\mu_n, \mu_\omega)$  of width  $2\Delta_n$  and height  $2\Delta_\omega$ , shifts the Heisenberg box to  $(N\mu_n, \mu_\omega/N)$  and scales its width to  $2N\Delta_n$  and its height to  $2\Delta_\omega/N$ . The repeated spectra in frequency appear because of upsampling.

will have a Heisenberg box of the same size, simply shifted by  $n_0$  and  $\omega_0$  (dashed boxes in Figure 6.11; we separated the effects of time and frequency shifts for clarity).

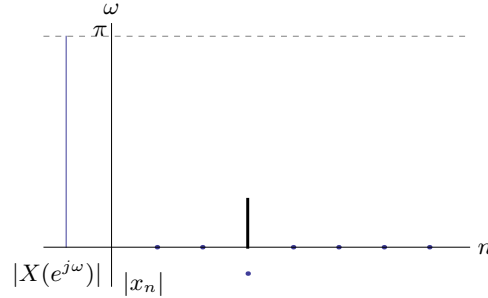
Similarly, a sequence obtained by upsampling of  $x_n$ ,

$$x(n/N) \xleftrightarrow{\text{DTFT}} X(e^{jN\omega}), \quad (6.17)$$

will have a Heisenberg box of the same area, scaled appropriately in time and frequency. That is, if  $x_n$  has a Heisenberg box specified by  $(\mu_n, \Delta_n, \mu_\omega, \Delta_\omega)$ , then the upsampled sequence has a Heisenberg box specified by  $(N\mu_n, N\Delta_n, \mu_\omega/N, \Delta_\omega/N)$  (dashed box in Figure 6.12). The effects of shift, modulation and scaling on the Heisenberg boxes are summarized in Table 6.2.

Sequence	Time center	Time spread	DTFT	Frequency center	Frequency spread
$x_n$	$\mu_n$	$\Delta_n$	$X(e^{j\omega})$	$\mu_\omega$	$\Delta_\omega$
$x_{n-n_0}$	$\mu_n + n_0$	$\Delta_n$	$e^{-j\omega n_0} X(e^{j\omega})$	$\mu_\omega$	$\Delta_\omega$
$e^{j\omega_0 n} x_n$	$\mu_n$	$\Delta_n$	$X(e^{j(\omega-\omega_0)})$	$\mu_\omega + \omega_0$	$\Delta_\omega$
bandlimited	$\mu_n/N$	$\Delta_n/N$	$\frac{1}{N} \sum_{k=0}^{N-1} X(e^{j(\frac{\omega-2\pi k}{N})})$	$N\mu_\omega$	$N\Delta_\omega$
upsampled & postfiltered	$N\mu_n$	$N\Delta_n$	$X(e^{jN\omega})$	$\mu_\omega/N$	$\Delta_\omega/N$

**Table 6.2:** Effect of a shift in time and frequency, as well as upsampling followed by postfiltering and downsampling of a bandlimited sequence in time and frequency, on a Heisenberg box  $(\mu_n, \Delta_n, \mu_\omega, \Delta_\omega)$ .



**Figure 6.13:** The Heisenberg box for the Kronecker delta sequence.

**EXAMPLE 6.6 (HEISENBERG BOX FOR THE KRONECKER DELTA SEQUENCE)** The Kronecker delta sequence from (2.7) is interesting to discuss as it seems to possess perfect time localization; in fact, both  $\mu_n = 0$  and  $\Delta_n^2 = 0$ . This means that the Heisenberg box is not bounded (no width); this does not violate Theorem 6.6 as it requires  $X(e^{j\pi}) = 0$ , clearly not satisfied by the DTFT of the Kronecker delta sequence, since it is constant everywhere,  $X(e^{j\omega}) = 1$ . Thus, this sequence is not a strictly lowpass sequence as we mentioned at the beginning of this section, a fact that has consequences in how we visualize its time-frequency localization properties using Heisenberg boxes. We can still draw its Heisenberg box, bearing in mind that it will be a line. For  $x_n = \delta_{n-3}$ , the Heisenberg 4-tuple is  $(3, 0, 0, \pi^2/3)$ ; Figure 6.13 shows the corresponding Heisenberg box.

**Uncertainty Principle** With these definitions paralleling those for continuous-time functions, we can obtain a result very similar to Theorem 6.3. One could imagine that it follows from combining Theorem 6.3 with Nyquist-rate sampling of a bandlimited function. The proof is shown in Solved Exercise 6.3 and suggests the use of the Cauchy–Schwarz inequality similarly to our earlier proof.

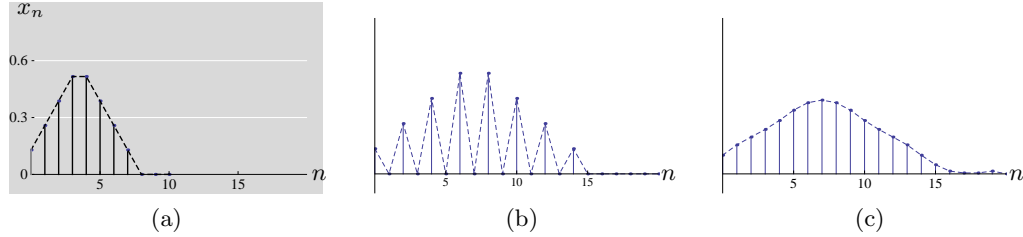
**THEOREM 6.6 (UNCERTAINTY PRINCIPLE FOR SEQUENCES)** Let  $x \in \ell^2(\mathbb{Z})$  with the DTFT  $X(e^{j\omega})$  and  $X(e^{j\pi}) = 0$ , the time spread  $\Delta_n$  and frequency spread  $\Delta_\omega$ . Then,

$$\Delta_n^2 \Delta_\omega^2 > \frac{1}{4}. \quad (6.18)$$

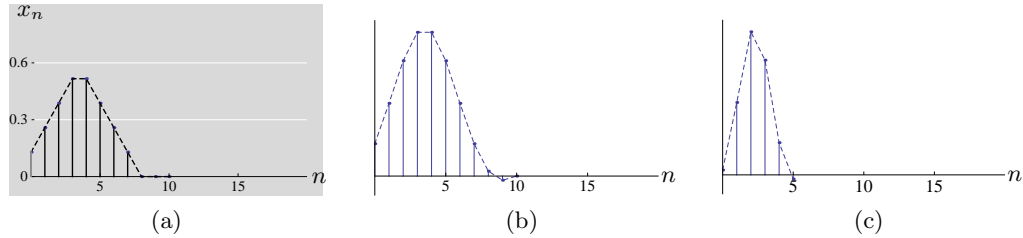
A standard example illustrating the tension between time and frequency localization is the analysis of a sequence containing a Kronecker delta sequence in time and a Dirac delta function in frequency (complex sinusoid/exponential):

$$x_n = \delta_{n-n_0} + e^{j\omega_0 n} \xleftrightarrow{\text{DTFT}} X(e^{j\omega}) = e^{-j\omega n_0} + 2\pi\delta(\omega - \omega_0). \quad (6.19)$$

Clearly, to locate the Kronecker delta sequence in time or the Dirac delta function in frequency, one needs to be as sharp as possible in that particular domain, thus



**Figure 6.14:** Stretching of a sequence by a factor 2. Envelopes in each case are drawn, to emphasize the fact that stretching is in fact upsampling followed by postfiltering as in Figure 4.13(b) with  $g$  an ideal halfband filter. (a) Original sequence. (b) Result after upsampling by 2. (c) Result after filtering with an ideal halfband filter.



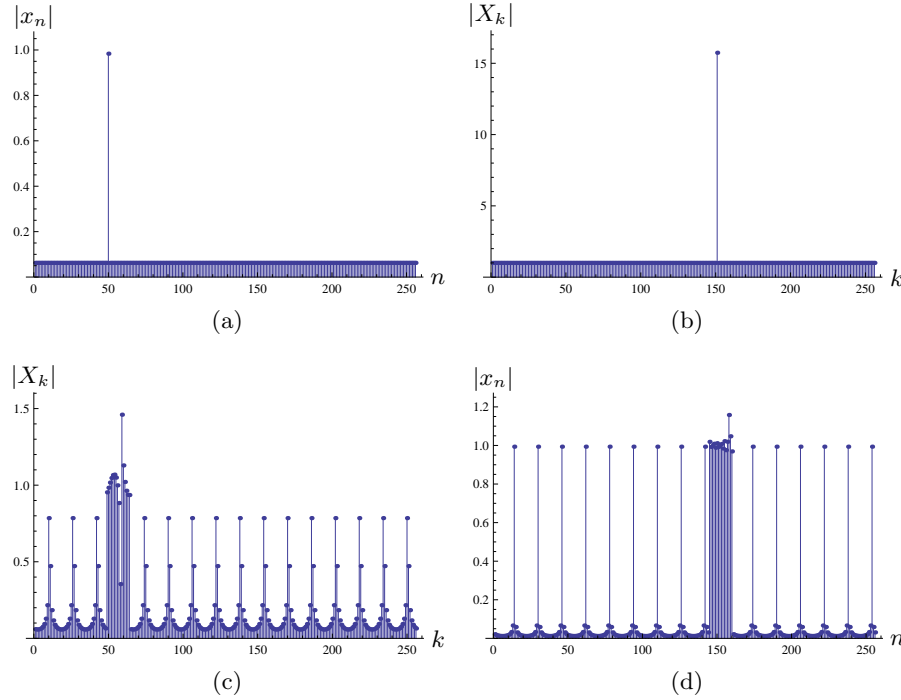
**Figure 6.15:** Contraction of a sequence by a factor 2. Envelopes in each case are drawn, to emphasize the fact that contraction is in fact prefiltering followed by downsampling as in Figure 4.13(a) with  $g$  an ideal halfband filter. (a) Original sequence. (b) Result after filtering with an ideal halfband filter. (c) Result after downsampling by 2.

compromising the sharpness in the other domain. We illustrate this for periodic sequences and their DFTs in Example 6.7.

### 6.3.4 Scale Localization

Unlike for functions, the notion of scaling is not as natural. For example, downsampling by a factor of  $N$  is in general a lossy operation, while upsampling by a factor of  $N$  introduces spurious zeros. What we will consider natural scaling operations in discrete domain are sampling (lowpass prefiltering followed by downsampling) and interpolation (upsampling followed by lowpass postfiltering) operations we introduced in Chapter 4. These are illustrated in Figures 6.14 and 6.15.

Thus, scale changes are more complicated in discrete time. In particular, compressing the time axis cannot be undone since samples are lost in the process. Scale changes by rational factors are possible through combinations of integer upsampling and downsampling, but cannot be undone in general (see Exercise 6.6).



**Figure 6.16:** Time-frequency resolution trade-off for a finite-length sequence consisting a Kronecker delta sequence in time and frequency. (a) Magnitude of the original sequence of length 256,  $x_n$ . (b) Magnitude response of its length-256 DFT,  $X_k$ . (c) Magnitude responses of the length-16 DFTs of the original sequence split into 16 pieces of length 16 each. (d) Magnitudes of the length-16 inverse DFTs of the original DFT split into 16 pieces of length 16 each.

### 6.3.5 Uncertainty Principle for Finite-Length Sequences

In addition to the uncertainty principle for infinite sequences, there exists a simple and powerful uncertainty principle for finite-length sequences and their DFTs.

#### THEOREM 6.7 (UNCERTAINTY PRINCIPLE FOR FINITE-LENGTH SEQUENCES)

Let  $x_n \in \mathbb{C}^N$  with the DFT  $X_k$ . Let  $N_n$  and  $N_k$  denote the number of nonzero components of  $x$  and  $X$ , respectively. Then,

$$N_n N_k \geq N. \quad (6.20)$$

It turns out that  $X$  cannot have  $N_n$  consecutive zeros (mod  $N$ ). This result is explored in Exercise 6.4 and arises again in Chapter 13.

**EXAMPLE 6.7 (A KRONECKER DELTA SEQUENCE IN TIME AND FREQUENCY)** We illustrate the fundamental trade-off between time and frequency localization with a numerical example. Consider a finite-length sequence of length  $N = 256$ , containing a Kronecker delta sequence in time and frequency, as shown in Figure 6.16(a). In Figure 6.16(b), we show the magnitude of its length-256 DFT, which has 256 frequency bins, and perfectly identifies the exponential component, while missing the time-domain Kronecker delta completely. To increase the time resolution, we divide the sequence into 16 pieces of length 16 each, taking the length-16 DFT of each, as shown in Figure 6.16(c). Now we can identify approximately where the time-domain Kronecker delta impulse occurs; however, the frequency resolution is reduced, since we now have only 16 frequency bins. Finally, Figure 6.16(d) shows the dual case, that is, we plot the magnitudes of length-16 inverse DFTs of the original DFT split into 16 pieces of length 16 each. Now we can identify approximately where the frequency-domain Kronecker delta impulse occurs; however, the time resolution is reduced, since we now have only 16 time bins.



## Chapter at a Glance

---

### Uncertainty Principle for Functions

---

For a function  $x(t) \in \mathcal{L}^2(\mathbb{R})$  with the Fourier transform  $X(\omega)$

energy in time	$\ x\ ^2$	$\int_{-\infty}^{\infty}  x(t) ^2 dt$
time center	$\mu_t$	$\frac{1}{\ x\ ^2} \int_{-\infty}^{\infty} t  x(t) ^2 dt$
time spread	$\Delta_t$	$\left( \frac{1}{\ x\ ^2} \int_{-\infty}^{\infty} (t - \mu_t)^2  x(t) ^2 dt \right)^{1/2}$
energy in frequency	$2\pi \ x\ ^2$	$\int_{-\infty}^{\infty}  X(\omega) ^2 d\omega$
frequency center	$\mu_\omega$	$\frac{1}{2\pi \ x\ ^2} \int_{-\infty}^{\infty} \omega  X(\omega) ^2 d\omega$
frequency spread	$\Delta_\omega$	$\left( \frac{1}{2\pi \ x\ ^2} \int_{-\infty}^{\infty} (\omega - \mu_\omega)^2  X(\omega) ^2 d\omega \right)^{1/2}$

---

then  $\Delta_t^2 \Delta_\omega^2 \geq 1/4$ , with equality achieved by a Gaussian  $x(t)$ .

---

### Uncertainty Principle for Sequences

---

For a sequence  $x_n \in \ell^2(\mathbb{Z})$  with the DTFT  $X(e^{j\omega})$

energy in time	$\ x\ ^2$	$\sum_{n \in \mathbb{Z}}  x_n ^2$
time center	$\mu_n$	$\frac{1}{\ x\ ^2} \sum_{n \in \mathbb{Z}} n  x_n ^2$
time spread	$\Delta_n$	$\left( \frac{1}{\ x\ ^2} \sum_{n \in \mathbb{Z}} (n - \mu_n)^2  x_n ^2 \right)^{1/2}$
energy in frequency	$2\pi \ x\ ^2$	$\int_{-\pi}^{\pi}  X(\omega) ^2 d\omega$
frequency center	$\mu_\omega$	$\frac{1}{2\pi \ x\ ^2} \int_{-\pi}^{\pi} \omega  X(\omega) ^2 d\omega$
frequency spread	$\Delta_\omega$	$\left( \frac{1}{2\pi \ x\ ^2} \int_{-\pi}^{\pi} (\omega - \mu_\omega)^2  X(\omega) ^2 d\omega \right)^{1/2}$

---

then  $\Delta_n^2 \Delta_\omega^2 > 1/4$ .

---

### Uncertainty Principle for Finite-Length Sequences

---

For a sequence  $x_n \in \mathbb{C}^N$  with the DFT  $X_k$

number of nonzero components of $x$	$N_n$
number of nonzero components of $X$	$N_\omega$

---

then  $N_n N_\omega \geq N$ .

---

## Historical Remarks



Uncertainty principles stemming from the Cauchy–Schwarz inequality have a long and rich history. The best known one is Heisenberg’s uncertainty principle in quantum physics, first developed in a 1927 essay [70]. **Werner Karl Heisenberg (1901–1976)** was a German physicist, credited as a founder of quantum mechanics, for which he was awarded the Nobel Prize in 1932. He had seven children, one of whom, Martin Heisenberg, was a celebrated geneticist. He collaborated with Bohr, Pauli and Dirac, among others. While he was initially attacked by the Nazi war machine for promoting Einstein’s views, he did head the Nazi nuclear project during the war. His role in the project has been a subject of controversy every since, with differing views on whether he was deliberately stalling Hitler’s efforts or not.

Kennard is credited with the first mathematically exact formulation of the uncertainty principle, and Robertson and Schrödinger provided generalizations. The uncertainty principle presented in Theorem 6.3 was proven by Weyl and Pauli and introduced to signal processing by **Dennis Gabor (1900–1979)** [55], a Hungarian physicist, and another winner of the Nobel Prize for physics (he is also known as inventor of holography). By finding a lower bound to  $\Delta_t \Delta_\omega$ , Gabor was intending to define an information measure or capacity for signals. Shannon’s communication theory [130] proved much more fruitful for this purpose, but Gabor’s proposal of signal analysis by shifted and modulated Gaussian functions has been a cornerstone of time-frequency analysis ever since. Slepian’s survey [133] is enlightening on these topics.



## Further Reading

Many of the uncertainty principles for discrete-time signals are considerably more complicated than Theorem 6.6. We have given only a result that follows papers by Ishii and Furukawa [79] and Calvez and Vilbé [22].

Donoho and Stark [46] derived new uncertainty principles in various domains. Particularly influential was an uncertainty principle for finite-dimensional signals and a demonstration of its significance for signal recovery (see Exercises 6.4 and 6.7). Moreover, Donoho and Huo [45] introduced performance guarantees for  $\ell^1$  minimization-based signal recovery algorithms; this has sparked a large body of work.

---

## Exercises with Solutions

### 6.1. Shift-Invariant Subspaces and Degrees of Freedom

Show that a function in the shift-invariant space  $S$  of piecewise-constant functions (4.1) but over intervals of length  $T > 0$  has exactly  $1/T$  degrees of freedom per unit time.

*Solution:* Consider an interval  $(-KT, KT)$  for some positive integer  $K$ . A function in  $S$

is specified on this interval by the  $2K$  inner products  $\{\langle x, \varphi_k \rangle\}_{k=-K}^{K-1}$ . Thus, the function has  $2K/(2KT) = 1/T$  degrees of freedom per unit time. The result follows from using a similar argument for an interval with length not necessarily an integer multiple of  $T$ .

### 6.2. Properties of Time and Frequency Spreads for Functions

Consider the time spread  $\Delta_t$  and the frequency spread  $\Delta_\omega$  defined in (6.2b) and (6.4b), respectively.

- (i) Show that the time shift and frequency shift (complex modulation) of  $x(t)$  as in (6.6) leave  $\Delta_t$  and  $\Delta_\omega$  unchanged.
- (ii) Show that the energy-conserving scaling of  $x(t)$  as in (6.7) scales  $\Delta_t$  by  $1/\alpha$ , while scaling  $\Delta_\omega$  by  $\alpha$ , thus leaving the time-frequency product unchanged.

*Solution:* Without loss of generality assume  $\|x\|^2 = 1$ .

- (i) The time shift  $y(t) = x(t - t_0)$ , or,  $Y(\omega) = e^{j\omega t_0} X(\omega)$  in frequency domain, changes  $\mu_{x,t}$  as follows:

$$\begin{aligned} \mu_{y,t} &= \int_{-\infty}^{\infty} t |y(t)|^2 dt = \int_{-\infty}^{\infty} t |x(t - t_0)|^2 dt \\ &\stackrel{(a)}{=} \int_{-\infty}^{\infty} (\tau + t_0) |x(\tau)|^2 d\tau \\ &= \int_{-\infty}^{\infty} \tau |x(\tau)|^2 d\tau + t_0 \int_{-\infty}^{\infty} |x(\tau)|^2 d\tau \stackrel{(b)}{=} \mu_{x,t} + t_0, \end{aligned}$$

where (a) follows from  $\tau = t - t_0$ ; and (b) from  $\|x\|^2 = 1$ .

The time spread  $\Delta_t^2$ , however, remains unchanged:

$$\begin{aligned} \Delta_{y,t}^2 &= \int_{-\infty}^{\infty} (t - \mu_{y,t})^2 |y(t)|^2 dt \\ &= \int_{-\infty}^{\infty} (t - \mu_{x,t} - t_0)^2 |x(t - t_0)|^2 dt \\ &\stackrel{(a)}{=} \int_{-\infty}^{\infty} (\tau - \mu_{x,t})^2 |x(\tau)|^2 d\tau = \Delta_{x,t}^2, \end{aligned}$$

where (a) again follows from  $\tau = t - t_0$ . Since  $|Y(\omega)|^2 = |X(\omega)|^2$ , the frequency spread is clearly not changed by time shift.

Similarly, the frequency shift  $Y(\omega) = X(\omega - \omega_0)$ , or,  $y(t) = e^{-j\omega_0 t} x(t)$ , changes  $\mu_{x,\omega}$  as follows:

$$\begin{aligned} \mu_{y,\omega} &= \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} \omega |Y(\omega)|^2 d\omega = \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} \omega |X(\omega - \omega_0)|^2 d\omega \\ &\stackrel{(a)}{=} \frac{1}{2\pi} \int_{w \in \mathbb{R}} (w + \omega_0) |X(w)|^2 dw \\ &= \frac{1}{2\pi} \int_{w \in \mathbb{R}} w |X(w)|^2 dw + \omega_0 \frac{1}{2\pi} \int_{w \in \mathbb{R}} |X(w)|^2 dw \stackrel{(b)}{=} \mu_{x,\omega} + \omega_0, \end{aligned}$$

where (a) follows from  $w = \omega - \omega_0$ ; and (b) from  $\|X(\omega)\| = 2\pi$  by Parseval's equality.

The frequency spread  $\Delta_{x,\omega}^2$ , however, remains unchanged:

$$\begin{aligned} \Delta_{y,\omega}^2 &= \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} (\omega - \mu_{y,\omega})^2 |Y(\omega)|^2 d\omega \\ &= \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} (\omega - \mu_{x,\omega} - \omega_0)^2 |X(\omega - \omega_0)|^2 d\omega \\ &\stackrel{(a)}{=} \frac{1}{2\pi} \int_{w \in \mathbb{R}} (w - \mu_{x,\omega})^2 |X(w)|^2 dw = \Delta_{x,\omega}^2, \end{aligned}$$

where (a) again follows from  $w = \omega - \omega_0$ . Similarly to the time shift not changing the frequency spread, the frequency shift does not change the time spread.

- (ii) Scaling  $y(t) = \sqrt{\alpha} x(\alpha t)$ , or,  $Y(\omega) = (1/\sqrt{\alpha}) X(\omega/\alpha)$  in frequency domain, changes  $\mu_{x,t}$  as follows:

$$\begin{aligned}\mu_{y,t} &= \int_{-\infty}^{\infty} t |y(t)|^2 dt = \int_{-\infty}^{\infty} t |\sqrt{\alpha} x(\alpha t)|^2 dt \\ &\stackrel{(a)}{=} \frac{1}{\alpha} \int_{-\infty}^{\infty} \tau |x(\tau)|^2 d\tau = \frac{1}{\alpha} \mu_{x,t},\end{aligned}$$

where (a) follows from  $\tau = \alpha t$ .

The time spread changes as well:

$$\begin{aligned}\Delta_{y,t}^2 &= \int_{-\infty}^{\infty} (t - \mu_{y,t})^2 |y(t)|^2 dt \\ &= \int_{-\infty}^{\infty} (t - \frac{1}{\alpha} \mu_{x,t})^2 |\sqrt{\alpha} x(\alpha t)|^2 dt \\ &\stackrel{(a)}{=} \frac{1}{\alpha^2} \int_{-\infty}^{\infty} (\tau - \mu_{x,t})^2 |x(\tau)|^2 d\tau = \frac{1}{\alpha^2} \Delta_{x,t}^2.\end{aligned}$$

where (a) again follows from  $\tau = \alpha t$ .

Scaling  $Y(\omega) = (1/\sqrt{\alpha}) X(\omega/\alpha)$ , or,  $y(t) = \sqrt{\alpha} x(\alpha t)$  in time domain, changes  $\mu_{x,\omega}$  as follows:

$$\begin{aligned}\mu_{y,\omega} &= \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} \omega |Y(\omega)|^2 d\omega = \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} \omega \left| \frac{1}{\sqrt{\alpha}} X\left(\frac{\omega}{\alpha}\right) \right|^2 d\omega \\ &\stackrel{(a)}{=} \frac{\alpha}{2\pi} \int_{w \in \mathbb{R}} w |X(w)|^2 dw = \alpha \mu_{x,\omega},\end{aligned}$$

where (a) follows from  $w = \omega/\alpha$ .

The frequency spread changes as well:

$$\begin{aligned}\Delta_{y,\omega}^2 &= \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} (\omega - \mu_{y,\omega})^2 |Y(\omega)|^2 d\omega \\ &= \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} (\omega - \alpha \mu_{x,\omega})^2 \left| \frac{1}{\sqrt{\alpha}} X\left(\frac{\omega}{\alpha}\right) \right|^2 d\omega \\ &= \frac{\alpha^2}{2\pi} \int_{w \in \mathbb{R}} (w - \mu_{x,\omega})^2 |X(w)|^2 dw = \alpha^2 \Delta_{x,\omega}^2,\end{aligned}$$

where (a) again follows from  $w = \omega/\alpha$ .

### 6.3. Uncertainty Principle for Sequences

Prove Theorem 6.6 for real sequences. Do not forget to provide an argument for the strictness of inequality (6.18).

(Hint: Use the Cauchy–Schwarz inequality (1.24) to bound  $\left| \int_{-\pi}^{\pi} \omega X(e^{j\omega}) \left[ \frac{\partial}{\partial \omega} X(e^{j\omega}) \right] d\omega \right|^2$ ).

*Solution:* Without loss of generality assume  $\|x\| = 1$ . Since  $x_n$  is real, according to Table 3.2,  $|X(e^{j\omega})|$  is even, so  $\mu_{\omega} = 0$ . We thus express the frequency spread from (6.14b) and the time spread from (6.12b) as

$$2\pi \Delta_{\omega}^2 = \int_{-\pi}^{\pi} |\omega X(e^{j\omega})|^2 d\omega, \quad (\text{E6.3-1a})$$

$$\begin{aligned}2\pi \Delta_n^2 &= 2\pi \sum_n (n - \mu_n)^2 x_n^2 = 2\pi \sum_n |-j(n - \mu_n)x_n|^2 \\ &\stackrel{(a)}{=} \int_{-\pi}^{\pi} \left| \frac{dX(e^{j\omega})}{d\omega} + j\mu_n X(e^{j\omega}) \right|^2 d\omega, \quad (\text{E6.3-1b})\end{aligned}$$

where (a) follows from Parseval's equality (2.103) and the differentiation-in-frequency property of the DTFT, (2.90).

Now let

$$\alpha = \left\langle \omega X(e^{j\omega}), \frac{dX(e^{j\omega})}{d\omega} + j\mu_n X(e^{j\omega}) \right\rangle \quad (\text{E6.3-2a})$$

$$\begin{aligned} &= \int_{-\pi}^{\pi} \omega X(e^{j\omega}) \frac{dX^*(e^{j\omega})}{d\omega} d\omega - j\mu_n \int_{-\pi}^{\pi} \omega |X(e^{j\omega})|^2 d\omega \\ &\stackrel{(a)}{=} \int_{-\pi}^{\pi} \omega X(e^{j\omega}) \frac{dX^*(e^{j\omega})}{d\omega} d\omega, \end{aligned} \quad (\text{E6.3-2b})$$

where in (a) the second term  $\int_{-\pi}^{\pi} \omega |X(e^{j\omega})|^2 d\omega$  is 0 because  $|X(e^{j\omega})|$  is even. Then,

$$\begin{aligned} 2\Re(\alpha) &= \alpha + \alpha^* = \int_{-\pi}^{\pi} \omega \left( X(e^{j\omega}) \frac{dX^*(e^{j\omega})}{d\omega} + X^*(e^{j\omega}) \frac{dX(e^{j\omega})}{d\omega} \right) d\omega \\ &= \int_{-\pi}^{\pi} \omega \frac{d}{d\omega} (X(e^{j\omega}) X^*(e^{j\omega})) d\omega = \int_{-\pi}^{\pi} \omega \frac{d}{d\omega} |X(e^{j\omega})|^2 d\omega \\ &\stackrel{(a)}{=} \omega |X(e^{j\omega})|^2 \Big|_{-\pi}^{\pi} - \int_{-\pi}^{\pi} |X(e^{j\omega})|^2 d\omega = -2\pi, \end{aligned} \quad (\text{E6.3-2c})$$

where (a) follows from integration by parts.

We can now use (E6.3-2c) to write

$$\begin{aligned} 4\pi^2 &= |-2\pi|^2 = 4|\Re(\alpha)|^2 \stackrel{(a)}{\leq} 4|\alpha|^2 \\ &\stackrel{(b)}{=} 4 \left| \left\langle \omega X(e^{j\omega}), \frac{dX(e^{j\omega})}{d\omega} + j\mu_n X(e^{j\omega}) \right\rangle \right|^2 \\ &\stackrel{(c)}{\leq} 4 \int_{-\pi}^{\pi} |\omega X(e^{j\omega})|^2 d\omega \int_{-\pi}^{\pi} \left| \frac{dX(e^{j\omega})}{d\omega} + j\mu_n X(e^{j\omega}) \right|^2 d\omega \\ &\stackrel{(d)}{=} 4(2\pi \Delta_{\omega}^2)(2\pi \Delta_n^2) = 4\pi^2 4 \Delta_{\omega}^2 \Delta_n^2, \end{aligned}$$

where (a) follows because for any  $\alpha \in \mathbb{C}$ ,  $|\Re(\alpha)| \leq |\alpha|$ ; (b) from (E6.3-2a); (c) from Cauchy–Schwarz inequality (1.24); and (d) from (E6.3-1a)–(E6.3-1b), proving the theorem. The Cauchy–Schwarz inequality holds with equality only with  $\beta \omega X(e^{j\omega}) = dX(e^{j\omega})/d\omega + j\mu_n X(e^{j\omega})$ , yielding a Gaussian function. As a Gaussian function is never zero, this contradicts the theorem’s condition that  $X(e^{j\pi}) = 0$ ; the theorem thus holds with strict inequality.

#### 6.4. Uncertainty Principle for Finite-Length Sequences

Let  $x_n \in \mathbb{C}^N$  with the DFT  $X_k$ . Let  $N_n$  and  $N_k$  denote the number of nonzero components of  $x$  and  $X$ , respectively.

- (i) Prove that  $X$  cannot have  $N_n$  consecutive zeros, where *consecutive* is interpreted mod  $N$ .  
(Hint: For an arbitrary selection of  $N_n$  consecutive components of  $X$ , form a linear system relating the nonzero components of  $x$  to the selected components of  $X$ .)
- (ii) Using (i), prove (6.20), the uncertainty principle for finite-length sequences, due to Donoho and Stark [46].

*Solution:*

- (i) Let  $i_0, i_1, \dots, i_{N_n-1}$  be the indices of the  $N_n$  nonzero components of  $x$ . Denote by  $y_n = x_{i_n}$  and by  $z_n = W_N^{i_n}$  for  $0 \leq n \leq N_n - 1$ .

We prove the result by contradiction. Suppose there exist  $N_n$  consecutive zero components of the DFT of  $x$ ,  $X_{k+m}$  for  $0 \leq m \leq N_n - 1$ , for some  $0 \leq k \leq N - 1$ . Observe that

$$X_{k+m} = \sum_{n=0}^{N-1} x_n W_N^{n(k+m)} = \sum_{n=0}^{N_n-1} y_n z_n^{(k+m)},$$

for any  $0 \leq m \leq N_n - 1$ . This system of linear equations can be expressed as

$$\hat{X} = \begin{bmatrix} X_k \\ \vdots \\ X_{k+N_n-1} \end{bmatrix} = Z \begin{bmatrix} y_0 \\ \vdots \\ y_{N_n-1} \end{bmatrix} = ZY,$$

where  $Z$  is an  $N_n \times N_n$  matrix with elements  $Z_{m,n} = z_n^{k+n}$ ,  $0 \leq m, n \leq N_n - 1$ .

By assumption,  $\hat{X}$  is a zero vector. Vector  $Y$  consists of nonzero components; hence, the matrix  $Z$  has a nontrivial null-space. Since  $Z$  is a square matrix, it must be rank-deficient, that is, its rank is smaller than  $N_n$ .

However,  $Z = \hat{Z}D$ , where  $\hat{Z}_{m,n} = z_n^m$ ,  $0 \leq m, n \leq N_n - 1$ ; and  $D = \text{diag}(z_n^{-k})_{0 \leq n \leq N_n-1}$ . The matrix  $\hat{Z}$  is a Vandermonde matrix as in (1.230) constructed from nonzero components; hence, it is of full rank  $N_n$ . The matrix  $D$  is a diagonal matrix with nonzero diagonal elements; hence, it is also of full rank  $N_n$ . As a product of two full-rank matrices,  $Z$  must also be a full-rank matrix, contradicting our previous statement. Thus, the  $X$  cannot have  $N_n$  consecutive zero components.

- (ii) Arrange the points of  $X$  in a circle and choose one nonzero component to start from. Because of (i), this nonzero component can be followed by at most  $N_n$  nonzero ones. Continuing the argument until we reach the initial point, we will have at least  $\lfloor N/N_n \rfloor + 1$  nonzero components. Thus, the total number of nonzero components will be

$$N_k \geq \left\lfloor \frac{N}{N_n} \right\rfloor + 1 \geq \frac{N}{N_n} \quad \Rightarrow \quad N_n N_k \geq N.$$

## Exercises

### 6.1. Time Spreads for B-Splines

Find the time spreads of the following B-splines:

- (i) Linear spline as in (5.33).
- (ii)  $N$ th order B-spline as in (5.31b).

### 6.2. Uncertainty Principle for Complex Functions

Prove Theorem 6.3 without assuming that  $x(t)$  is a real function.

(Hint: The proof requires more than the Cauchy-Schwarz inequality and integration by parts. Use the product rule of differentiation,  $(|x(t)|^2)' = x'(t)x^*(t) + x(t)(x^*(t))'$ . Also, use that for any  $\alpha \in \mathbb{C}$ ,  $|\alpha| \geq |\alpha + \alpha^*|/2$ .)

### 6.3. Properties of Modified Time and Frequency Spreads for Functions

(i)-(ii) from Solved Exercise 6.2 hold for the time-frequency spreads  $\Delta_t^{(\beta)}$  and  $\Delta_\omega^{(\beta)}$  defined in (6.10a) and (6.10b), respectively.

### 6.4. Sequences with Finite Number of Nonzero Terms and Their DTFTs

Show that if a sequence has a finite number of nonzero terms, then its DTFT cannot be zero over an interval (that is, it can only have isolated zeros). Conversely, show that if a DTFT is zero over an interval, then the corresponding sequence has an infinite number of nonzero terms.

### 6.5. Properties of Time and Frequency Spreads for Sequences

Consider the time spread  $\Delta_n$  and the frequency spread  $\Delta_\omega$  defined in (6.12b) and (6.14b), respectively.

- (i) Show that the time shift and frequency shift (complex modulation) of  $x_n$  leave  $\Delta_n$  and  $\Delta_\omega$  unchanged.
- (ii) Show that the upsampling by  $N$  followed by ideal lowpass postfiltering of  $x_n$  scales  $\Delta_n$  by  $N$ , while scaling  $\Delta_\omega$  by  $1/N$ , thus leaving the time-frequency product un-

changed. Show the counterpart of the same result for downsampling of a bandlimited sequence  $x \in \text{BL}[-\pi/N, \pi/N]$ .

6.6. *Rational Scale Changes for Sequences*

A scale change by a factor  $M/N$  can be achieved by upsampling by  $M$  followed by downsampling by  $N$ .

- (i) Consider a scale change by  $3/2$ , and show that it can be implemented either by upsampling by 3, followed by downsampling by 2, or the converse.
- (ii) Let  $M$  and  $N$  be coprime. Show that a sampling rate change by  $M/N$  cannot be undone unless  $N = 1$ .

6.7. *Signal Recovery Based on the Uncertainty Principle for Finite-Length Sequences*

The DFT  $X_k$  of a length- $N$  sequence  $x_n$  is known to have only  $N_k$  nonzero components. Using the result of Solved Exercise 6.4, show that this limited DFT-domain support allows for a recovery of a unique  $x_n$  from any  $M$  time-domain components provided that

$$2(N - M) N_k < N.$$

(Hint: Show that nonunique recovery leads to a contradiction.)





# Intermezzo

## Bridging Parts I and II

We are about to embark on Part II of the book, where more advanced topics and more complex tools will be derived, all based on the foundations laid down in Part I of the book. This chapter serves as an intermezzo<sup>103</sup> between the two parts.

### Tools

Where do we stand after the coverage of Part I? We have seen a number of basic and powerful tools and concepts:

**Geometry** The inner product in the Hilbert spaces we consider leads to a familiar, yet powerful, geometrical view of signals and spaces. They include orthogonality between signals (vectors), and best approximation on a subspace by orthogonal projections.

**Existence of Bases** The fact that (separable) Hilbert spaces allow for bases leads to natural representations in orthonormal or biorthogonal bases. In the case of linear operators, the basis of eigenvectors is particularly attractive.

**Fourier Representations** For the signals and systems we consider, systems operate on signals using the convolution operator; this naturally leads to various forms of the Fourier transform. In particular, the eigensequence/eigenfunction property of complex exponentials lead naturally to linear shift-invariant systems.

**Sampling and Interpolation** The interaction between the discrete and the continuous world is realized using sampling and interpolation, two powerful tools that connect these two worlds.

**Approximation and Compression** When signals cannot be perfectly represented, approximations are necessary. Then, quantitative results on how close an approximated or compressed version is to the original, are very useful.

---

<sup>103</sup>*Intermezzo* is a short connecting instrumental movement in a musical piece.

## Adapting Tools to Real-World Problems

These basic results form the foundations upon which to build tools that are not only more advanced but also more practical; these tools need to be adapted to real-world problems taking into account the following:

**Finiteness and Localization** Any real-world signal is of finite duration, although we use infinite ones, such as sine waves, as a convenient mathematical abstraction. Moreover, real-world signals are typically transient. How do we localize our signal analysis? How do we trade off localization (sharpness) in one domain for localization in the other, for example time and frequency domains? How do we detect transients?

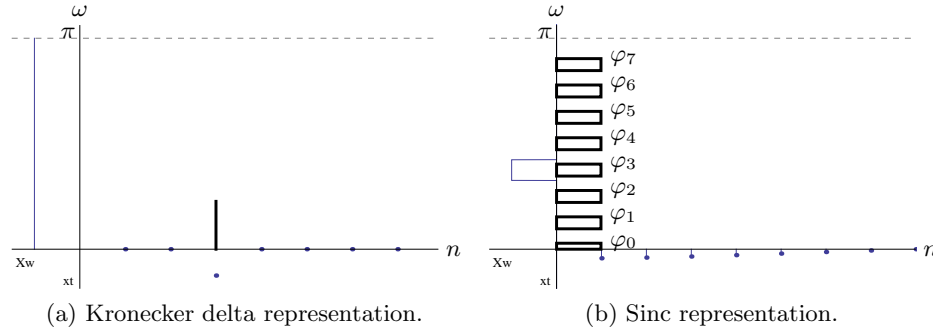
**Prior Knowledge** It helps to know what we are looking for: more often than not, we have some prior information about the signal or event we are interested in. Such priors on the signal class or the noise will help shape the solution.

**Limitations** There are limits on what can be done: in the linear measurement case, bounds such as the uncertainty principle set limits to how sharp an analysis can ever be. In a noisy setting, estimation theory provides lower bounds on the variance of an estimator. For compression and communication, information theory bounds the performance of any possible scheme, by rate-distortion and capacity regions. Such bounds are useful in at least two fundamental ways: (1) they separate what can be done from what is impossible; and (2) they provide yardsticks for comparing practical systems to the theoretical performance limits.

**Computational Aspects** Whatever the solution, it needs to be computable; for example, some of constructive solutions providing bounds on performance lead to hopelessly complex algorithms. Even seemingly simple problems such as best approximation in a frame require exhaustive search, and thus become impractical for real-world problems. Instead, we will seek approximate solutions together with performance bounds. Often, the problem is structured and can thus lead to savings in computation. For problems that are shift invariant or can be decomposed into pieces that are, FFT can be used. However, since real-world signals are of finite, but arbitrary length, no algorithm can use an FFT over the entire signal, since it would not have linear complexity. Thus, time localization is necessary from the computational point of view as well.

## Bases and the Time-Frequency Plane

To visualize localization properties of a representation, we use the time-frequency plane and Heisenberg boxes, conceptual tools we introduced in the last chapter. For example, Figure 6.3 showed a Heisenberg box plotted in the time-frequency plane for a function  $x(t)$  with a Fourier transform  $X(\omega)$ , conveying the idea that there exists a center of mass described by the function's center in time and frequency  $(\mu_t, \mu_\omega)$ , with a spread in time and frequency  $(\Delta_t, \Delta_\omega)$ . While this particular figure



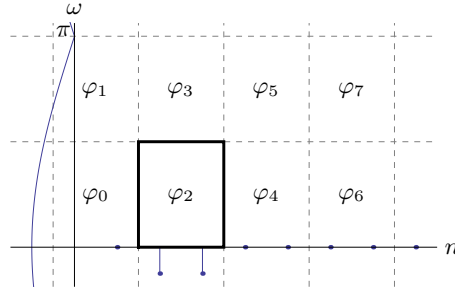
**Figure I.1:** Two representations with extreme localization properties. (a) The Dirac representation has perfect localization in time and no localization in frequency. (b) The sinc representation has perfect localization in frequency and no localization in time.

showed a Heisenberg box for a single function, one can draw these for all functions in a basis, for example, yielding an abstract time-frequency representation. This time-frequency representation is often called a spectrogram, and is used in a variety of applications, from speech and music analysis to mode detection and denoising.

As an example, let us consider two extremes of representing a finite-length sequence: the standard basis consisting of shifted Kronecker delta impulses, and the sinc basis. From what we have seen in Figure 6.16, we expect that the inner products with Kronecker delta basis sequences will isolate time-local events, while the inner products with the inverse sinc basis sequences will isolate frequency-local events; each representation has perfect localization properties in one domain and no localization in the other. This is illustrated in Figure I.1, where thick boxes are the Heisenberg boxes for the basis function  $\varphi_3$  in each case. For the Kronecker delta basis, the extent of the basis function covers the entire frequency axis (no frequency localization) and just the immediate neighborhood of the Kronecker delta impulse (perfect time localization). In contrast, for the sinc basis, the extent of the basis function covers the entire time axis (no time localization) and just the immediate neighborhood of the box in frequency (perfect frequency localization).

**Haar: An Efficient Orthonormal Basis with Time-Frequency Structure** As a simple case study, we now build an orthonormal basis for  $\ell^2(\mathbb{Z})$ , but with some localization in both domains. For example, starting from the Kronecker delta basis, can we improve its localization in frequency slightly while not trading too much of its time locality? Figure I.2 illustrates the desired tiling, where each tile from Figure I.1(a) has been divided in two in frequency, thereby improving frequency localization; the price we pay is slightly worse time localization, where each tile has become twice as wide.

Given this tiling, can we now find sequences to produce such a tiling? Let us concentrate first on the lower left-hand tile with the basis function  $\varphi_0$ . We will search for the simplest  $\varphi_0$  which has roughly the time spread of 2 and frequency



**Figure I.2:** Desired time-frequency tiling for an orthonormal basis with a slightly better frequency localization than the Kronecker delta basis, but at a price of a slightly worse time localization.

spread of  $\pi/2$ . Assume we ask for  $\varphi_0$  to be exactly of length 2, that is,

$$\varphi_{0,n} = \cos \theta \delta_n + \sin \theta \delta_{n-1}, \quad (\text{I.1})$$

where we have also imposed  $\|\varphi_0\| = 1$ , as we want it to be a part of an orthonormal basis. From (6.12a), the time center for such a  $\varphi_0$  would be  $\mu_n = \sin^2 \theta$ , and from (6.12b), its time spread would be  $\delta_n = \sin^2(2\theta)/4$ . How about its frequency behavior? We are looking for  $\varphi_0$  to be roughly a halfband lowpass sequence; let us thus ask for it to block the highest frequency  $\pi$ :

$$\Phi_0(e^{j\omega})|_{\omega=\pi} = \cos \theta + \sin \theta e^{-j\omega}|_{\omega=\pi} = \cos \theta - \sin \theta = 0. \quad (\text{I.2})$$

Solving the above equation yields  $\theta = k\pi + \pi/4$  and

$$\varphi_{0,n} = \frac{1}{\sqrt{2}}(\delta_n + \delta_{n-1}) \quad \varphi_0 = \left[ \dots \quad 0 \quad \boxed{\frac{1}{\sqrt{2}}} \quad \frac{1}{\sqrt{2}} \quad 0 \quad 0 \quad \dots \right]^T, \quad (\text{I.3})$$

with the time center  $\mu_n = 1/2$  and time spread  $\delta_n = 1/2$ . We now repeat the process and try to find a  $\varphi_1$  of length 2 being a roughly halfband highpass sequence. We can use (I.1) as a general form of a sequence of length 2 and norm 1. What we look for now is a sequence which has to be orthogonal to the first candidate basis vector  $\varphi_0$  (if they are to be a part of an orthonormal basis), that is

$$\langle \varphi_0, \varphi_1 \rangle = \frac{1}{\sqrt{2}} \cos \theta + \frac{1}{\sqrt{2}} \sin \theta = 0, \quad (\text{I.4})$$

which yields  $\theta = (2k+1)\pi/2 + \pi/4$ , and one possible form of  $\varphi_1$ :

$$\varphi_{1,n} = \frac{1}{\sqrt{2}}(\delta_n - \delta_{n-1}) \quad \varphi_1 = \left[ \dots \quad 0 \quad \boxed{\frac{1}{\sqrt{2}}} \quad -\frac{1}{\sqrt{2}} \quad 0 \quad 0 \quad \dots \right]^T. \quad (\text{I.5})$$

Note that while we did not specifically impose it, the resulting sequence is indeed highpass in nature, as  $\Phi_1(e^{j\omega})|_{\omega=0} = 0$ .

So far so good; we only have infinitely many more functions to find. To make the task less daunting, we search for an easier way, for example, by shifting  $\varphi_0$  and  $\varphi_1$  along the time axis by integer multiples of 2. Call  $\varphi_{2k,n} = \varphi_{0,n-2k}$  and  $\varphi_{2k+1,n} = \varphi_{1,n-2k}$ . Then indeed, the set  $\Phi = \{\varphi_k\}_{k \in \mathbb{Z}}$  is an orthonormal set, which:

- (i) possesses structure in terms of time and frequency localization properties (it serves as an almost perfect localization tool in time, and a rather rough one in frequency);
- (ii) is efficient (it is built from two template functions and their shifts).

As we well know from Chapter 1, orthonormality of a set is not enough for that set to form an orthonormal basis; it must be complete as well. We now show that this set is true. An easy way to do this is to observe that  $\varphi_{2k}$  and  $\varphi_{2k+1}$  each operate on two samples of the input sequence only: for  $n = 2k, 2k+1$ . Thus, it is enough to show that  $\varphi_{2k}, \varphi_{2k+1}$  form an orthonormal basis for those  $x \in \ell^2(\mathbb{Z})$  which are nonzero only for  $n = 2k, 2k+1$ . This is further equivalent to showing that vectors  $(1/\sqrt{2}) [1 \ 1]^T$  and  $(1/\sqrt{2}) [1 \ -1]^T$  form an orthonormal basis for  $\mathbb{R}^2$ , a trivial fact. As this argument holds for all  $k$ , we indeed have that  $\Phi$  is an orthonormal basis for  $\ell^2(\mathbb{Z})$ , known under the name *Haar basis*:

$$\begin{aligned}
 x &= \sum_{k \in \mathbb{Z}} \langle x, \varphi_k \rangle \varphi_k \\
 &= \sum_{k \in \mathbb{Z}} \langle x, \varphi_{2k} \rangle \varphi_{2k} + \sum_{k \in \mathbb{Z}} \langle x, \varphi_{2k+1} \rangle \varphi_{2k+1} \\
 &= \underbrace{\sum_{k \in \mathbb{Z}} \frac{1}{\sqrt{2}} (x_{2k} + x_{2k+1}) \varphi_{2k}}_{x_V} + \underbrace{\sum_{k \in \mathbb{Z}} \frac{1}{\sqrt{2}} (x_{2k} - x_{2k+1}) \varphi_{2k+1}}_{x_W}, \tag{I.6}
 \end{aligned}$$

where we have separated the basis functions into two groups: those which are obtained by shifting the lowpass template  $\varphi_0$ , and those which are obtained by shifting the highpass template  $\varphi_1$ .

From the same argument we just went through, we see what would happen if we would remove one tile/basis function, say  $\varphi_{2k}$ : those sequences

We can now use all the machinery we developed in Chapter 1, and look at projections, matrix view, etc. For example, the matrix representing this basis is

$$\Phi = \frac{1}{\sqrt{2}} \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & 0 & 0 & 0 & 0 & \dots \\ \dots & \boxed{1} & 1 & 0 & 0 & \dots \\ \dots & 1 & -1 & 0 & 0 & \dots \\ \dots & 0 & 0 & 1 & 1 & \dots \\ \dots & 0 & 0 & 1 & -1 & \dots \\ \dots & 0 & 0 & 0 & 0 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \ddots \\ & \underbrace{\phantom{\vdots}}_{\varphi_0} & \underbrace{\phantom{\vdots}}_{\varphi_1} & \underbrace{\phantom{\vdots}}_{\varphi_2} & \underbrace{\phantom{\vdots}}_{\varphi_3} & \end{bmatrix}. \tag{I.7}$$

The matrix is block diagonal, with blocks of size  $2 \times 2$ .

From (I.6), we can immediately see that the Haar orthonormal basis projects onto two subspaces: lowpass space  $V$ , spanned by the lowpass template  $\varphi_0$  and its even shifts, and the highpass space  $W$ , spanned by the highpass template  $\varphi_1$  and its even shifts:

$$V = \text{span}(\{\varphi_{0,n-2k}\}_{k \in \mathbb{Z}}), \quad W = \text{span}(\{\varphi_{1,n-2k}\}_{k \in \mathbb{Z}}). \quad (\text{I.8})$$

From (I.6), the lowpass and highpass projections are:

$$x_V = \left[ \dots \begin{bmatrix} \frac{1}{2}(x_0 + x_1) \end{bmatrix} \quad \frac{1}{2}(x_0 + x_1) \quad \frac{1}{2}(x_2 + x_3) \quad \frac{1}{2}(x_2 + x_3) \quad \dots \right]^T \quad (\text{I.9a})$$

$$x_W = \left[ \dots \begin{bmatrix} \frac{1}{2}(x_0 - x_1) \end{bmatrix} \quad -\frac{1}{2}(x_0 - x_1) \quad \frac{1}{2}(x_2 - x_3) \quad -\frac{1}{2}(x_2 - x_3) \quad \dots \right]^T \quad (\text{I.9b})$$

Indeed,  $x_V$  is a *smoothed* version of  $x$  where every two samples have been replaced by their average, while  $x_W$  is the *detailed* version of  $x$  where every two samples have been replaced by their difference (and its negative). As the sequences in  $V$  and  $W$  are orthogonal, and the expansion is a basis,

$$\ell^2(\mathbb{Z}) = V \oplus W. \quad (\text{I.10})$$

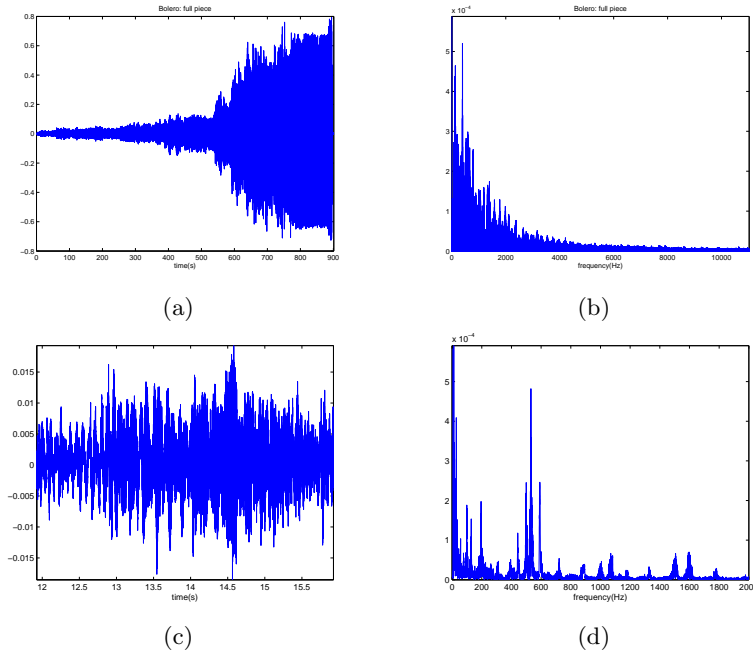
In the next chapter, we will show efficient ways of implementing this, and other, more general orthonormal bases, using filter banks.

## Examples of Real-World Signals

To motivate the construction of specific tools in Part II, we now show a few real-world signals with associated problems, and broadly outline possible solutions.

**The Transient Nature of Polyphonic Music** We start with one of the most famous and popular pieces of western classical music, Ravel's Bolero. It starts with a single instrument, the piccolo flute, almost whispering the theme, and ends with the full, 120-instrument orchestra thundering the finale. Clearly, the local characteristics of the piece evolve dramatically from beginning to end, illustrated by the 15-minute time-domain display of the acoustical signal in Figure I.3(a). While the Fourier transform of that sequence, shown in Figure I.3(b), exhibits a number of spectral peaks corresponding to some key harmonic structures, the signature evolution of the Bolero from the introduction to the grand finale is lost.

To understand the local behavior, we look at a short piece, from 12 – 15 sec of the Bolero and its Fourier transform. Figure I.3(c) shows the local behavior in time, while Figure I.3(d) shows the local behavior in frequency, that is, those frequencies that are active in this part of the piece. We immediately come across a practical issue: extracting a part of the Bolero means multiplying the time-domain sequence with a rectangular window as was done in Example 2.4. In frequency domain, this windowing process amounts to a convolution with the Fourier transform of the window, smoothing the spectrum. This is the cost we pay for localizing Bolero in time.



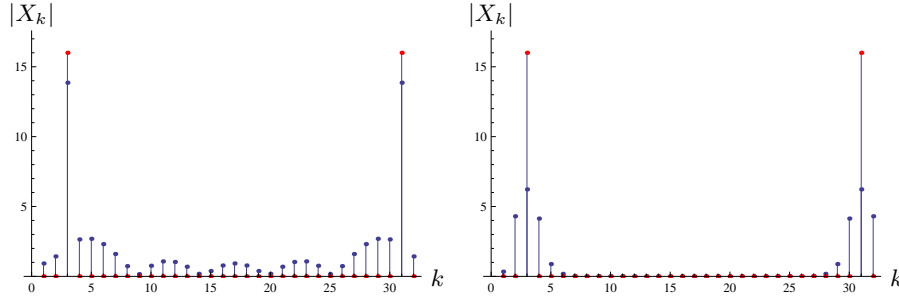
**Figure I.3:** The entire Ravel's Bolero in (a) time and (b) frequency domains. A 4-sec segment in (c) time and (d) frequency domains.

We have seen this effect in Example 2.4 in time domain only. Figure I.4 shows it for a sinusoid and two different windows from Example 2.4 both in time and in frequency. With no windowing, the DFT contains a double peak (because the sequence is real, red stems in the figure). With a sharp window in time, rectangular window (2.12), the DFT is the convolution of the two peaks with the sinc in frequency domain, creating a diluted version of the peaks as in Figure I.4(a). With a smoother window in time, raised cosine window (2.15), the DFT is sharper and closer to peaks as in Figure I.4(a).

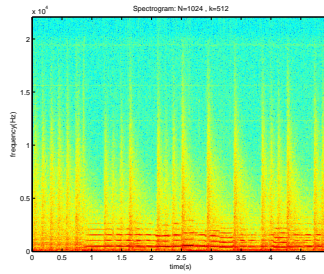
Windowing leads us to notion of a local, or, short-time Fourier transform, introduced in Chapter 8. The idea is to apply a moving window over the signal, followed by a Fourier transform. The resulting local Fourier spectrum has now two indices; one for the location of the window, or a time index, and one for the local frequency index.

This discussions leads us naturally to the analysis in the time-frequency plane, shown in Figure 6.3. The frequency analysis is now localized around a given time location. A time-frequency plot of Ravel's Bolero is shown in Figure I.5. Now, one sees clearly the evolution of the piece in this time segment, both in terms of spectral evolution, rhythms and intensity.

The idea of time-frequency analysis is also central to the MP3 audio-coding standard, where it is performed using an orthonormal time-frequency basis (which



**Figure I.4:** The DFT of a sinusoidal sequence  $x_n = \sin((\pi/8)n + \pi/2)$  (red stems) from Figure 2.2 and the DFTs of its windowed versions  $w_n x_n$  using (a) the rectangular window (2.12) and (b) the raised cosine window (2.15), both of length  $n_0 = 26$  (stem plots).



**Figure I.5:** Time-frequency analysis of a segment of Ravel's Bolero.

we will construct basis in Chapter 8). Following this analysis, the expansion coefficients in the orthonormal basis are carefully quantized so as to minimize perceptual distortion. Overall, the bit rate can be reduced by an almost order of magnitude without audible distortion.

**Visions of an Image** In our music-analysis example, we mapped a one-dimensional sequence into a two-dimensional one, an image. Starting with an image, we might end up with an even higher-dimensional representation. To explore this, we look again at *All is Vanity* shown in Figure 6.2(a).

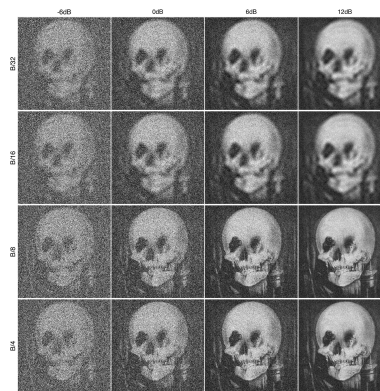
One very standard representation, often used in computer vision, is the image pyramid. It consists in computing a lowpass approximation (or projection onto the space of lowpass images) as well as a difference image. The latter is an approximation to a derivative of the image, thus enhancing edges, while being close to zero in smooth parts. The block diagram of one step of such a decomposition is shown in Figure I.6(a), while the usual projection and difference is displayed in Figure I.6(b). Of course one can iterate this scheme. Instead of showing the projections, one usually shows the coefficients, that is the downsampled images, both lowpass and



## TfBD

**Figure I.6:** Pyramid decomposition. (a) Block diagram. (b) Geometry of decomposition.

## TfBD

**Figure I.7:** Pyramid decomposition of “All is Vanity”. (a) Sequence of lowpass and downsampled images. (b) Respective difference images.**Figure I.8:** The notion of resolution. (a)-(d) Original as well as lowpass version. (e)-(h) Original as well as noisy version.

difference. This is shown in Figure I.7, where we see recursively the successive low-pass filtered and downsampled images, as well as the difference between successive levels. As is clear from the sequence of lowpass versions, the “lady in front of the mirror” soon gives way to a skull... Because the pyramid decomposition can be perfectly inverted (as will be shown in Chapter 10), the difference images contain all it takes to go from the skull to a young lady. A word of care is necessary here: typically, the lowpass version is a good predictor for the full-resolution image. This is clearly not the case here, since the intent of the painter was to create a visual illusion, or a perceptual tension between the sharp and the blurred versions of the image. Lowpass filtering, or reducing bandwidth, can be seen as loss of resolution. A similar effect can be achieved by reducing the signal to noise ratio, or adding noise. Perceptually, we will have trouble identifying the relevant information because of noise. Figure I.8 compares reducing bandwidth with adding noise, indicating the similarity of the two effects. Intuitively, the notion of “resolution” is therefore related to the ability to extract information. In this case, the underlying information is the “lady in front of the mirror”, and too much noise or insufficient bandwidth both preclude our ability to extract this information. The pyramid scheme seen above is both simple and intuitive. Its only drawback is that it is redundant, for

## TfBD

**Figure I.9:** Wavelet decomposition (a) Coefficients in the various channels (b) Projection onto the various subspaces.

example, one step as in Figure I.6(a) maps  $N$  samples into  $N/2$  lowpass coefficients and  $N$  difference samples, an increase by 50% (in two dimensions, the increase is smaller, since the lowpass coefficients amount to  $N^2/4$  for  $N^2$  input pixels).

If one chooses a wavelet decomposition instead, it is possible to have an orthonormal basis and expansion therein. The details will be the topic of Chapter 9, suffices to say that the lowpass version is similar to the pyramid case, while the difference channel (in two dimensions) is now made up of three channels with the combinations of highpass/lowpass in horizontal and vertical directions using separable two-dimensional filters. Since each channel is downsampled by 4, we have no redundancy.

Note: Do we put a block diagram?

Figure I.9 shows a wavelet decomposition of “All is Vanity”, where the decomposition is iterated on the smoothed, lowpass version. In part (a), the coefficients are shown, while in part (b), the projections onto the various subspaces are depicted. Similarly to the pyramid case seen earlier, the lady vanishes to leave a skull as we go down the repeated lowpass branch, or the projection onto the lowpass restriction. The various highpass branches enhance vertical, horizontal, or diagonal edges, depending on which highpass filter is involved. This wavelet decomposition of images is at the heart of the JPEG2000 image compression standard. In addition to the decomposition, or analysis, the compression involves sophisticated quantization and bit allocation to reach compression factors by an order of magnitude without much visible artifacts. For reconstruction, since it is an orthonormal decomposition, one simply uses the transposed operator, or upsampling and interpolation.

If there is such a thing as JPEG2000, there must be a straight JPEG as a precursor. Indeed, a simpler image coding standard preceded the one based on wavelets, and its simplicity and good performance<sup>104</sup> keeps JPEG as the front runner image coding standard. Interestingly, it is in some sense closer to a short-time Fourier analysis than to a wavelet decomposition. To make things short, an image is sliced up into blocks of  $M$  by  $M$  pixels, and each block is transformed by a two-dimensional DCT, which is similar to a real valued discrete Fourier transform. In the DCT domain, smart quantization and entropy coding reaches good compression.

Since cutting into blocks is like applying a window of size  $M$  by  $M$ , and taking a DCT similar to a DFT, we have indeed a short-time Fourier transform like analysis. Figure I.10 shows a DCT analysis of our now classic image in (a), the individually transformed blocks in (b), and the reordered coefficients such that coefficients of the same frequency indices  $(i,j)$  from each block are gathered together. Sure enough, the skull appears clearly in the coefficients of order  $(0,0)$ , or local averages in each block.

<sup>104</sup>As well as a clearer situation regarding the patent situation.

## TfBD

**Figure I.10:** Transform used in JPEG (a) Original (b) DCT coefficients. (c) Reordered DCT coefficients.

## TfBD

**Figure I.11:** Communication or signaling over a channel. Based on a sequence  $\{\alpha_i\}$ , a signal is synthesized to be sent over a channel, typically a convolution. At the receiver, estimates  $\{\hat{\alpha}_i\}$  of the coefficient are retrieved.

**Communications with the Short-Time Fourier Transform** The two previous examples were representation problems, or taking a signal and finding a good expansion space to achieve a more compact representation.

For communication, we look at a dual problem, namely signaling. We want to synthesize a signal based on some given coefficients  $\{\alpha_i\}$  that carry information. This synthesized signal is then sent over a channel, typically represented by a convolution, possibly slowly varying over time, and received at a decoder where the sequence  $\{\alpha_i\}$  or an approximation  $\{\hat{\alpha}_i\}$  needs to be retrieved. The situation is schematically shown in Figure I.11. If the channel is well modeled by a convolution with known impulse response  $h(t)$ , it is natural to use complex sinusoids for signaling. Since they are eigenfunctions of the convolution operator, the various  $\{\alpha_i\}$  are not mixed by the channel, but simply weighted by the Fourier transform of the impulse response, according to the Fourier convolution theorem, (3.64). For simplicity, we recall the simplest version in matrix algebra. Assume a vector  $x \in \mathbb{C}^N$  and a circular convolution matrix  $H$ . If we use  $x$  directly as input to the channel, we obtain as output

$$y = Hx,$$

or, a convolved version of the various input values. Instead, use the DFT basis vectors for signaling, each weighted by an entry of  $x$ , or,

$$X = Fx.$$

The effect of the channel is to weight the vectors of  $F$  by the DFT of filter,

$$Y = HX = HFx \stackrel{(a)}{=} F\Lambda x,$$

where (a) follow from ... and  $\Lambda$  is the diagonal matrix of DFT coefficients of the filter. Assuming  $\Lambda$  known and invertible (no zeros in the DFT spectrum), then we can retrieve the original values by inverse DFT and equalization of the channel,

$$\hat{x} = \Lambda^{-1}F^{-1}Y,$$

where  $\hat{x} = x$  in this idealized, noiseless case. This “signaling by Fourier basis vectors” is conceptually depicted in Figure I.12.

TfBD

**Figure I.12:** Block diagram of DFT signaling over a circular convolution channel.

TfBD

**Figure I.13:** Time-frequency signaling or modulation. Basis functions localized both in time and frequency are used to carry information over a slowly varying convolution channel.

Now, real life is more complicated than the abstracted version above, which assumed discrete-time, periodic sequences. The idea is to use localized Fourier basis vectors that live within a certain time window. This is again a short-time Fourier transform view of the problem. While these localized basis functions do not satisfy an exact eigenfunction property anymore, they do so approximately, and well enough to be able to carry information over a channel. In addition, if the channel is time varying, like in mobile communication, the finite time support of the basis functions become advantageous. That is, one can apply a local equalization to compensate for the channel effects and the time of the localized signaling, something long basis functions cannot do. In sum, time-frequency signaling achieves both time locality as well as an approximate eigenfunction property. Combined with a localized equalization to deal with time-varying channels, it permits efficient transmission over many communication channels. As a case in point, the orthogonal frequency division multiplexing (OFDM) standard is the basis for many successful communication methods, like for example Wi-Fi.

Fig I.13 shows schematically a time-frequency signaling method that underlies OFDM. Of course, a practical system involves many more components, like for example using pilots to estimate the channel, which are dedicated time frequency basis functions used to probe the time-varying channel.

**Predicting the Future Using Orthogonal Projections** Here something about LPC. TBD.

That's all, folks!

## Part II: Structured Representations for Signal Processing

The second part of the book develops a variety of signal representations and brings all the tools together leading to sparse signal processing. It covers both discrete- and continuous-time signals; sets of vectors that form bases and frames; and Fourier and wavelet constructions. We progress from discrete time in Chapters 7–10 to continuous time in Chapters 11–12, and, finally, to Chapter 13, which gives an introduction and overview of the main ingredients of sparse signal processing

Within these sets of chapters, we cover Fourier techniques first (Chapters 8 and 11) and wavelet techniques second (Chapters 9 and 12). Representations using frames (*oversampled* representations) exist in both discrete and continuous time and in both Fourier and wavelet style; these are covered in Chapters 10–12.

**Chapter 7: Filter Banks: Building Blocks of Time-Frequency Expansions**, develops two-channel filter banks as a computationally-efficient tool both for computing basis expansions of discrete-time signals as well as for the reconstruction of these signals. The representations associated with two-channel filter banks are simultaneously the simplest of Fourier and wavelet representations on  $\ell^2(\mathbb{Z})$ ; thus, the detailed developments in this chapter lay the groundwork for Chapters 8–10.

**Chapter 8: Local Fourier Bases on Sequences**, develops local Fourier representations for  $\ell^2(\mathbb{Z})$ , through a generalization from two channels to  $N$  channels. It yields local Fourier series representations of sequences, also known as windowed Fourier, Gabor, or short-time Fourier representations. They inherently split  $\ell^2(\mathbb{Z})$  into  $N$  bands equally spaced in frequency. While there exist no good local Fourier bases (apart from minimally short ones), there exist good *local cosine* bases, where the complex-exponential modulation is replaced by cosine modulation. Moreover, there exist good *local Fourier frames*, a topic covered in Chapter 10.

**Chapter 9: Wavelet Bases on Sequences**, develops wavelet representations for  $\ell^2(\mathbb{Z})$ , through a generalization from two channels to tree-structured filter banks. These tree-structured filter banks are not arbitrary, rather they build unbalanced trees by iterating on the lowpass (coarse) branch only. This results in an *octave-band* filter bank, or a *discrete wavelet transform*.

**Chapter 10: Local Fourier and Wavelet Frames on Sequences**, devel-

ops frame representations for  $\ell^2(\mathbb{Z})$ , both in Fourier and wavelet flavors, through a generalization from critically-sampled to oversampled filter banks. The redundancy inherent in the representation allows for significant freedom in design. Not only can we look for the *best expansion*, we can also look for the *best expansion coefficients* given a fixed expansion and under desired constraints (sparsity being one example).

**Chapter 11: Local Fourier Transforms, Frames and Bases on Functions** develops local Fourier representations for  $\mathcal{L}^2(\mathbb{R})$ . The aim follows that of Chapter 8, but for functions. We look for ways to localize the analysis Fourier transform provided by windowing the complex exponentials. The chapter starts with the most redundant representation, the local Fourier transform, and then samples it to obtain local Fourier frames. With critical sampling we then try for local Fourier bases, where, again, bases with simultaneously good time and frequency localization do not exist, a result known as the Balian-Low theorem. Cosine local Fourier bases do exist, as do wavelet ones we discuss in the next chapter.

**Chapter 12: Wavelet Bases, Frames and Transforms on Functions** develops wavelet representations for  $\mathcal{L}^2(\mathbb{R})$ . It shows how to overcome the roadblock from Chapter 11 when trying to construct local Fourier bases with reasonable joint time and frequency localization. While bases are possible with cosine, instead of complex-exponential, modulation, we can do even better. In this chapter, we start with a whole wealth of wavelet bases, and then go in the direction of increasing redundancy, by building frames and finally the continuous wavelet transform.

**Chapter 13: Approximation, Estimation, and Compression** develops the main ingredients of sparse signal processing, namely approximation methods and their performance, estimation procedures based on adapted representations, for example, denoising, compression methods and their performance, and finally a glimpse at inverse problems, including sparse sampling methods. As the aim of many signal processing algorithms is to transform high-dimensional problems into smaller-dimensional ones, and this by either linear or nonlinear methods, the basic axiom is that there exists a representation where the initial high-dimensional problem has a low-dimensional solution. This sweeping statement needs some qualifications, since such a solution is clearly not always known. However, we show many where there is a solution of this kind and of which we give a brief overview. The classic Karhunen-Loève expansion of stochastic processes is an example where linear approximation is optimal for least-squares approximation. The success of compression methods using the DCT and wavelets relies on nonlinear approximation in bases. And some recent solutions for inverse problems rely on a sparsity prior in a basis or a frame representation.

## Chapter 7

# Filter Banks: Building Blocks of Time-Frequency Expansions

## Contents

7.1	Introduction . . . . .	568
7.2	Orthogonal Two-Channel Filter Banks . . . . .	572
7.3	Design of Orthogonal Two-Channel Filter Banks	585
7.4	Biorthogonal Two-Channel Filter Banks . . . . .	592
7.5	Design of Biorthogonal Two-Channel Filter Banks	602
7.6	Two-Channel Filter Banks with Stochastic Inputs	607
7.7	Computational Aspects . . . . .	608
	Chapter at a Glance . . . . .	615
	Historical Remarks . . . . .	618
	Further Reading . . . . .	618
	Exercises with Solutions . . . . .	620
	Exercises . . . . .	625

The aim of this chapter is to build discrete-time bases with desirable time-frequency features and structure that enable tractable analysis and efficient algorithmic implementation. We achieve these goals by constructing bases via filter banks.

Using filter banks provides an easy way to understand the relationship between analysis and synthesis operators, while, at the same time, making their efficient implementation obvious. Moreover, filter banks are at the root of the constructions of wavelet bases in Chapters 9 and 12. In short, together with discrete-time filters and the FFT, filter banks are among the most basic tools of signal processing.

This chapter deals exclusively with two-channel filter banks since they are (1) the simplest; (2) reveal the essence of the  $N$ -channel ones; and (3) are used as building blocks for more general bases. We focus first on the orthogonal case, which is the most structured and has the easiest geometric interpretation. Due to its importance in practice, we follow with the discussion of the biorthogonal

case. We consider real-coefficient filter banks exclusively; pointers to complex-coefficient ones, as well as to various generalizations, such as  $N$ -channel filter banks, multidimensional filter banks and transmultiplexers, are given in *Further Reading*.

## 7.1 Introduction

### Implementing a Haar Orthonormal Basis Expansion

At the end of the previous chapter, we constructed an orthonormal basis for  $\ell^2(\mathbb{Z})$  which possesses structure in terms of time and frequency localization properties (it serves as an almost perfect localization tool in time, and a rather rough one in frequency); and, is efficient (it is built from two template sequences, one lowpass and the other highpass, and their shifts). This was the so-called Haar basis.

What we want to do now is implement that basis using signal processing machinery. We first rename our template basis sequences from (I.3) and (I.5) as:

$$g_n = \varphi_{0,n} = \frac{1}{\sqrt{2}}(\delta_n + \delta_{n-1}), \quad (7.1a)$$

$$h_n = \varphi_{1,n} = \frac{1}{\sqrt{2}}(\delta_n - \delta_{n-1}). \quad (7.1b)$$

This is done both for simplicity, as well as because it is the standard way these sequences are denoted. We start by rewriting the reconstruction formula (I.6) as

$$\begin{aligned} x_n &= \sum_{k \in \mathbb{Z}} \underbrace{\langle x, \varphi_{2k} \rangle}_{\alpha_k} \varphi_{2k,n} + \sum_{k \in \mathbb{Z}} \underbrace{\langle x, \varphi_{2k+1} \rangle}_{\beta_k} \varphi_{2k+1,n} \\ &= \sum_{k \in \mathbb{Z}} \alpha_k \underbrace{\varphi_{2k,n}}_{g_{n-2k}} + \sum_{k \in \mathbb{Z}} \beta_k \underbrace{\varphi_{2k+1,n}}_{h_{n-2k}} \\ &= \sum_{k \in \mathbb{Z}} \alpha_k g_{n-2k} + \sum_{k \in \mathbb{Z}} \beta_k h_{n-2k}, \end{aligned} \quad (7.2)$$

where we have renamed the basis functions as in (7.1), as well as denoted the expansion coefficients as

$$\langle x, \varphi_{2k} \rangle = \langle x_n, g_{n-2k} \rangle_n = \alpha_k, \quad (7.3a)$$

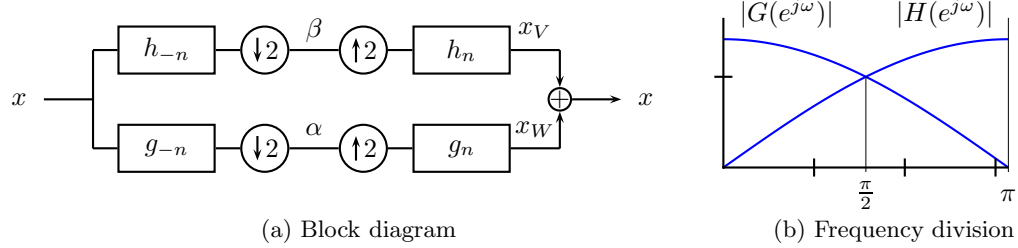
$$\langle x, \varphi_{2k+1} \rangle = \langle x_n, h_{n-2k} \rangle_n = \beta_k. \quad (7.3b)$$

Then, recognize each sum in (7.2) as the output of upsampling followed by filtering (2.198) with the input sequences being  $\alpha_k$  and  $\beta_k$ , respectively. Thus, the first sum in (7.2) can be implemented as the input sequence  $\alpha$  going through an upsampler by 2 followed by filtering by  $g$ , and the second as the input sequence  $\beta$  going through an upsampler by 2 followed by filtering by  $h$ .

By the same token, we can identify the computation of the expansion coefficients in (7.3) as (2.195), that is, both  $\alpha$  and  $\beta$  sequences can be obtained using filtering by  $g_{-n}$  followed by downsampling by 2 (for  $\alpha_k$ ), or filtering by  $h_{-n}$  followed by downsampling by 2 (for  $\beta_k$ ).

We can put together the above operations to yield a *two-channel filter bank* implementing a Haar orthonormal basis expansion as in Figure 7.1(a). The left part





**Figure 7.1:** A two-channel analysis/synthesis filter bank. (a) Block diagram, where an analysis filter bank is followed by a synthesis filter bank. In the orthogonal case, the impulse responses of the analysis filters are time-reversed versions of the impulse responses of the synthesis filters. The filter  $g$  is typically lowpass, while the filter  $h$  is typically highpass. (b) Frequency responses of the two Haar filters computing averages and differences, showing the decomposition into low- and high-frequency content.

that computes the expansion coefficients is termed an *analysis filter bank*, while the right part that computes the projections is termed a *synthesis filter bank*.

As before, once we have identified all the appropriate multirate components, we can examine the Haar filter bank via matrix operations (linear operators). For example, in matrix notation, the analysis process (7.3) can be expressed as

$$\begin{bmatrix} \vdots \\ \boxed{\alpha_0} \\ \beta_0 \\ \alpha_1 \\ \beta_1 \\ \alpha_2 \\ \beta_2 \\ \vdots \end{bmatrix} = \frac{1}{\sqrt{2}} \underbrace{\begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & \boxed{1} & 1 & 0 & 0 & 0 & 0 & \dots \\ \dots & 1 & -1 & 0 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 1 & 1 & 0 & 0 & \dots \\ \dots & 0 & 0 & 1 & -1 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & 0 & 1 & 1 & \dots \\ \dots & 0 & 0 & 0 & 0 & 1 & -1 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{\Phi^T} \begin{bmatrix} \vdots \\ \boxed{x_0} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ \vdots \end{bmatrix}, \quad (7.4)$$

and the synthesis process (7.2) as

$$\underbrace{\begin{bmatrix} \vdots \\ \boxed{x_0} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ \vdots \end{bmatrix}}_x = \frac{1}{\sqrt{2}} \underbrace{\begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & \boxed{1} & 1 & 0 & 0 & 0 & 0 & \dots \\ \dots & 1 & -1 & 0 & 0 & 0 & 0 & \dots \\ \dots & 0 & 0 & 1 & 1 & 0 & 0 & \dots \\ \dots & 0 & 0 & 1 & -1 & 0 & 0 & \dots \\ \dots & 0 & 0 & 0 & 0 & 1 & 1 & \dots \\ \dots & 0 & 0 & 0 & 0 & 1 & -1 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{\Phi} \underbrace{\begin{bmatrix} \vdots \\ \boxed{\alpha_0} \\ \beta_0 \\ \alpha_1 \\ \beta_1 \\ \alpha_2 \\ \beta_2 \\ \vdots \end{bmatrix}}_X, \quad (7.5)$$

or

$$x = \Phi \Phi^T x \quad \Rightarrow \quad \Phi \Phi^T = I. \quad (7.6)$$

Of course, the matrix  $\Phi$  is the same matrix we have seen in (I.7). Moreover, from (7.6), it is a unitary matrix, which we know from Chapter 1, *Chapter at a Glance*, implies that the Haar basis is an orthonormal basis (and have already shown in Chapter 6). Table 7.8 gives a summary of the Haar filter bank in various domains.

### Implementing a General Orthonormal Basis Expansion

What we have seen for the Haar orthonormal basis is true in general; we can construct an orthonormal basis for  $\ell^2(\mathbb{Z})$  using two template basis sequences and their even shifts. As we have seen, we can implement such an orthonormal basis using a two-channel filter bank, consisting of downsamplers, upsamplers and filters  $g$  and  $h$ . Let  $g$  and  $h$  be two real-coefficient, causal filters,<sup>105</sup> where we implicitly assume that these filters have certain time and frequency localization properties, as discussed in Chapter 6 ( $g$  is lowpass and  $h$  is highpass). The synthesis (7.5) generalizes to

$$\begin{bmatrix} \vdots \\ x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \end{bmatrix} = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & g_0 & h_0 & 0 & 0 & 0 & 0 & \dots \\ \dots & g_1 & h_1 & 0 & 0 & 0 & 0 & \dots \\ \dots & g_2 & h_2 & g_0 & h_0 & 0 & 0 & \dots \\ \dots & g_3 & h_3 & g_1 & h_1 & 0 & 0 & \dots \\ \dots & g_4 & h_4 & g_2 & h_2 & g_0 & h_0 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ \alpha_0 \\ \beta_0 \\ \alpha_1 \\ \beta_1 \\ \alpha_2 \\ \vdots \end{bmatrix} = \Phi X, \quad (7.7)$$

with the basis matrix  $\Phi$  as before. To have an orthonormal basis, the basis sequences  $\{\varphi_k\}_{k \in \mathbb{Z}}$ —even shifts of template sequences  $g$  and  $h$ , must form an orthonormal set as in (1.83), or,  $\Phi$  must be unitary, implying its columns are orthonormal:

$$\langle g_n, g_{n-2k} \rangle_n = \delta_k, \quad \langle h_n, h_{n-2k} \rangle_n = \delta_k, \quad \langle g_n, h_{n-2k} \rangle_n = 0. \quad (7.8)$$

We have seen in (2.209) that such filters are called orthogonal; how to design them is a central topic of this chapter.

As we are building an orthonormal basis, computing the expansion coefficients of an input sequence means taking the inner product between that sequence and each basis sequence. In terms of the orthonormal set given by the columns of  $\Phi$ ,

<sup>105</sup>While causality is not necessary to construct a filter bank, we impose it later and it improves readability here. We stress again that we deal exclusively with real-coefficient filters.

this amounts to a multiplication by  $\Phi^T$ :

$$\underbrace{\begin{bmatrix} \vdots \\ \boxed{\alpha_0} \\ \beta_0 \\ \alpha_1 \\ \beta_1 \\ \alpha_2 \\ \vdots \end{bmatrix}}_X = \underbrace{\begin{bmatrix} \vdots \\ \boxed{\langle x_n, g_n \rangle_n} \\ \langle x_n, h_n \rangle_n \\ \langle x_n, g_{n-2} \rangle_n \\ \langle x_n, h_{n-2} \rangle_n \\ \langle x_n, g_{n-4} \rangle_n \\ \vdots \end{bmatrix}}_X = \underbrace{\begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \cdots & \boxed{g_0} & g_1 & g_2 & g_3 & g_4 & \cdots \\ \cdots & h_0 & h_1 & h_2 & h_3 & h_4 & \cdots \\ \cdots & 0 & 0 & g_0 & g_1 & g_2 & \cdots \\ \cdots & 0 & 0 & h_0 & h_1 & h_2 & \cdots \\ \cdots & 0 & 0 & 0 & 0 & g_0 & \cdots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{\Phi^T} \underbrace{\begin{bmatrix} \vdots \\ \boxed{x_0} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \end{bmatrix}}_x. \quad (7.9)$$

As in the Haar case, this can be implemented with convolutions by  $g_{-n}$  and  $h_{-n}$ , followed by downsampling by 2—an analysis filter bank as in Figure 7.1(a). In filter bank terms, the representation of  $x$  in terms of a basis (or frame) is called *perfect reconstruction*.

Thus, what we have built is as in Chapter 6—an orthonormal basis with structure (time and frequency localization properties) as well as efficient implementation guaranteed by the filter bank. As in the Haar case, this structure is seen in the subspaces  $V$  and  $W$  on which the orthonormal basis projects; we implicitly assume that  $V$  is the space of coarse (lowpass) sequences and  $W$  is the space of detail (highpass) sequences. Figure 7.3 illustrates that, where a synthetic sequence with features at different scales is split into lowpass and highpass components. These subspaces are spanned by the lowpass template  $g$  and its even shifts ( $V$ ) and the highpass template  $h$  and its even shifts ( $W$ ) as in (I.8):

$$V = \overline{\text{span}}(\{\varphi_{0,n-2k}\}_{k \in \mathbb{Z}}) = \overline{\text{span}}(\{g_{n-2k}\}_{k \in \mathbb{Z}}), \quad (7.10a)$$

$$W = \overline{\text{span}}(\{\varphi_{1,n-2k}\}_{k \in \mathbb{Z}}) = \overline{\text{span}}(\{h_{n-2k}\}_{k \in \mathbb{Z}}), \quad (7.10b)$$

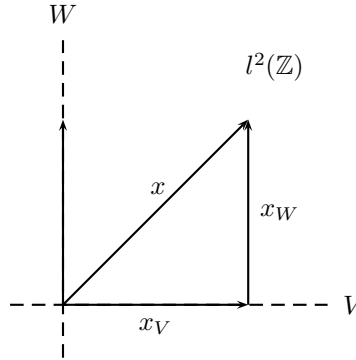
and produce the lowpass and highpass approximations, respectively:

$$x_V = \sum_{k \in \mathbb{Z}} \alpha_k g_{n-2k}, \quad (7.11a)$$

$$x_W = \sum_{k \in \mathbb{Z}} \beta_k h_{n-2k}. \quad (7.11b)$$

As the basis sequences spanning these spaces are orthogonal to each other and all together form an orthonormal basis, the two projection subspaces together give back the original space as in (I.10):  $\ell^2(\mathbb{Z}) = V \oplus W$ .

In this brief chapter preview, we introduced the two-channel filter bank as in Figure 7.1(a). It uses orthogonal filters satisfying (7.8) and computes an expansion with respect to the set of basis vectors  $\{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}$ , yielding a decomposition into approximation spaces  $V$  and  $W$  having complementary signal processing properties. Our task now is to find appropriate filters (template basis sequences) and develop properties of the filter bank in detail. We start by considering the lowpass filter  $g$ , since everything else will follow from there. We concentrate only on real-coefficient FIR filters since they are dominant in practice.



**Figure 7.2:** A sequence  $x$  is split into two approximation sequences  $x_V$  and  $x_W$ . An orthonormal filter bank ensures that  $x_V$  and  $x_W$  are orthogonal and sum up to the original sequence. We also show the split of  $\ell^2(\mathbb{Z})$  into two orthogonal complements  $V$  (lowpass subspace) and  $W$  (highpass subspace).

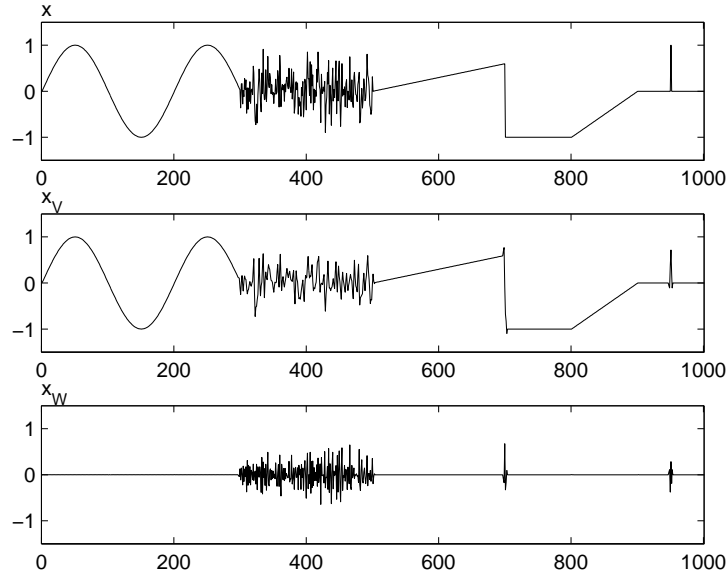
### Chapter Outline

We start by showing how orthonormal bases are implemented by orthogonal filter banks in Section 7.2 and follow by discussing three approaches to the design of orthogonal filter banks in Section 7.3. We then discuss the theory and design of biorthogonal filter banks in Sections 7.4 and 7.5. In Section 7.6, we discuss stochastic filter banks, followed by algorithms in Section 7.7.

*Notation used in this chapter:* In this chapter, we consider real-coefficient filter banks exclusively; pointers to complex-coefficient ones are given in *Further Reading*. Thus, Hermitian transposition will occur rarely; when filter coefficients are complex, the transposition in some places should be Hermitian transposition, however, only coefficients should be conjugated and not  $z$ . We will point these out throughout the chapter.  $\square$

## 7.2 Orthogonal Two-Channel Filter Banks

This section develops necessary conditions for the design of orthogonal two-channel filter banks implementing orthonormal bases and the key properties of such filter banks. We assume that the system shown in Figure 7.1(a) implements an orthonormal basis for sequences in  $\ell^2(\mathbb{Z})$  using the basis sequences  $\{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}$ . We first determine what this means for the lowpass and highpass channels separately, and follow by combining the channels. We then develop a polyphase representation for orthogonal filter banks and discuss their polynomial approximation properties.



**Figure 7.3:** A sequence and its projections. (a) The sequence  $x$  with different-scale features (low-frequency sinusoid, high-frequency noise, piecewise polynomial and a Kronecker delta sequence). (b) The lowpass projection  $x_V$ . (c) The highpass projection  $x_W$ .

### 7.2.1 A Single Channel and Its Properties

We now look at each channel of Figure 7.1 separately and determine their properties. As the lowpass and highpass channels are essentially symmetric, our approach is to establish (1) the properties inherent to each channel on its own; and (2) given one channel, establish the properties the other has to satisfy so as to build an orthonormal basis when combined. While we have seen most of the properties already, we summarize them here for completeness.

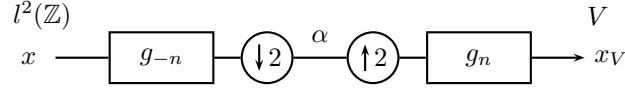
Consider the lower branch of Figure 7.1(a), projecting the input  $x$  onto its lowpass approximation  $x_V$ , depicted separately in Figure 7.4. In (7.11a), that lowpass approximation  $x_V$  was given as

$$x_V = \sum_{k \in \mathbb{Z}} \alpha_k g_{n-2k}. \quad (7.12a)$$

Similarly, in (7.11b), the highpass approximation  $x_W$  was given as

$$x_W = \sum_{k \in \mathbb{Z}} \beta_k h_{n-2k}. \quad (7.12b)$$

**Orthogonality of the Lowpass Filter** Since we started with an orthonormal basis, the set  $\{g_{n-2k}\}_{k \in \mathbb{Z}}$  is an orthonormal set. We have seen in Section 2.7.5 that such

**Figure 7.4:** The lowpass branch of a two-channel filter bank, mapping  $x$  to  $x_V$ .

a filter is termed orthogonal and satisfies (2.209):

$$\begin{aligned}
 \langle g_n, g_{n-2k} \rangle &= \delta_k & \begin{array}{c} \text{Matrix View} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_2 G^T G U_2 = I \\ G(z)G(z^{-1}) + G(-z)G(-z^{-1}) = 2 \\ |G(e^{j\omega})|^2 + |G(e^{j(\omega+\pi)})|^2 = 2 \end{array} \quad (7.13)
 \end{aligned}$$

In the matrix view, we have used linear operators (infinite matrices) introduced in Section 2.7. These are: (1) downsampling by 2,  $D_2$ , from (2.180a); (2) upsampling by 2,  $U_2$ , from (2.185a); and (3) filtering by  $G$ , from (2.63). The matrix view expresses the fact that the columns of  $GU_2$  form an orthonormal set.<sup>106</sup> The DTFT version is the quadrature mirror formula from (2.208).

---

#### Lowpass Channel in a Two-Channel Orthogonal Filter Bank

---

##### Lowpass filter

Original domain	$g_n$	$\langle g_n, g_{n-2k} \rangle_n = \delta_k$
Matrix domain	$G$	$D_2 G^T G U_2 = I$
$z$ -domain	$G(z)$	$G(z)G(z^{-1}) + G(-z)G(-z^{-1}) = 2$
DTFT domain	$G(e^{j\omega})$	$ G(e^{j\omega}) ^2 +  G(e^{j(\omega+\pi)}) ^2 = 2$ (quadrature mirror formula)
Polyphase domain	$G(z) = G_0(z^2) + z^{-1}G_1(z^2)$	$G_0(z)G_0(z^{-1}) + G_1(z)G_1(z^{-1}) = 1$

##### Deterministic autocorrelation

Original domain	$a_n = \langle g_k, g_{k+n} \rangle_k$	$a_{2k} = \delta_k$
Matrix domain	$A = G^T G$	$D_2 A U_2 = I$
$z$ -domain	$A(z) = G(z)G(z^{-1})$	$A(z) + A(-z) = 2$ $A(z) = 1 + 2 \sum_{k=0}^{\infty} a_{2k+1} (z^{2k+1} + z^{-(2k+1)})$
DTFT domain	$A(e^{j\omega}) =  G(e^{j\omega}) ^2$	$A(e^{j\omega}) + A(e^{j(\omega+\pi)}) = 2$

##### Orthogonal projection onto smooth space $V = \overline{\text{span}}(\{g_{n-2k}\}_{k \in \mathbb{Z}})$

$$x_V = P_V x \quad P_V = G U_2 D_2 G^T$$


---

**Table 7.1:** Properties of the lowpass channel in an orthogonal two-channel filter bank. Properties for the highpass channel are analogous.

<sup>106</sup>We remind the reader once more that we are considering exclusively real-coefficient filter banks, and thus transposition instead of Hermitian transposition in (7.13).

**Orthogonality of the Highpass Filter** Similarly to  $\{g_{n-2k}\}_{k \in \mathbb{Z}}$ , the set  $\{h_{n-2k}\}_{k \in \mathbb{Z}}$  is an orthonormal set, and the sequence  $h$  can be seen as the impulse response of an orthogonal filter satisfying:

$$\begin{array}{ccc} \langle h_n, h_{n-2k} \rangle = \delta_k & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_2 H^T H U_2 = I \\ H(z)H(z^{-1}) + H(-z)H(-z^{-1}) = 2 \\ |H(e^{j\omega})|^2 + |H(e^{j(\omega+\pi)})|^2 = 2 \end{array} \end{array} \quad (7.14)$$

The matrix view expresses the fact that the columns of  $HU_2$  form an orthonormal set. Again, the DTFT version is the quadrature mirror formula from (2.208).

**Deterministic Autocorrelation of the Lowpass Filter** As it is widely used in filter design, we rephrase (7.13) in terms of the deterministic autocorrelation of  $g$ , given by (2.96):

$$\begin{array}{ccc} \langle g_n, g_{n-2k} \rangle = a_{2k} = \delta_k & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_2 A U_2 = I \\ A(z) + A(-z) = 2 \\ A(e^{j\omega}) + A(e^{j(\omega+\pi)}) = 2 \end{array} \end{array} \quad (7.15)$$

In the above,  $A = G^T G$  is a symmetric matrix with element  $a_k$  on the  $k$ th diagonal left/right from the main diagonal. Thus, except for  $a_0$ , all the other even terms of  $a_k$  are 0, leading to

$$A(z) \stackrel{(a)}{=} G(z)G(z^{-1}) = 1 + 2 \sum_{k=0}^{\infty} a_{2k+1} (z^{2k+1} + z^{-(2k+1)}), \quad (7.16)$$

where (a) follows from (2.142).

**Deterministic Autocorrelation of the Highpass Filter** Similarly to the lowpass filter,

$$\begin{array}{ccc} \langle h_n, h_{n-2k} \rangle = a_{2k} = \delta_k & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_2 A U_2 = I \\ A(z) + A(-z) = 2 \\ A(e^{j\omega}) + A(e^{j(\omega+\pi)}) = 2 \end{array} \end{array} \quad (7.17)$$

Equation (7.16) holds for this deterministic autocorrelation as well.

**Orthogonal Projection Property of the Lowpass Channel** We now look at the lowpass channel as a composition of four linear operators we just saw:

$$x_V = P_V x = G U_2 D_2 G^T x. \quad (7.18)$$

The notation is evocative of projection onto  $V$ , and we will now show that the lowpass channel accomplishes precisely this. Using (7.13), we check idempotency

and self-adjointness of  $P$  (Definition 1.27),

$$\begin{aligned} P_V^2 &= (GU_2 \underbrace{D_2 G^T}_I) (GU_2 D_2 G^T) \stackrel{(a)}{=} GU_2 D_2 G^T = P_V, \\ P_V^T &= (GU_2 D_2 G^T)^T = G(U_2 D_2)^T G^T \stackrel{(b)}{=} GU_2 D_2 G^T = P_V, \end{aligned}$$

where (a) follows from (7.13) and (b) from (2.190). Indeed,  $P_V$  is an orthogonal projection operator, with the range given in (7.10a):

$$V = \overline{\text{span}}(\{g_{n-2k}\}_{k \in \mathbb{Z}}). \quad (7.19)$$

The summary of properties of the lowpass channel is given in Table 7.1.

**Orthogonal Projection Property of the Highpass Channel** The highpass channel as a composition of four linear operators (infinite matrices) is:

$$x_W = P_W x = H U_2 D_2 H^T x. \quad (7.20)$$

It is no surprise that  $P_W$  is an orthogonal projection operator with the range given in (7.10b):

$$W = \overline{\text{span}}(\{h_{n-2k}\}_{k \in \mathbb{Z}}). \quad (7.21)$$

The summary of properties of the highpass channel is given in Table 7.1 (table provided for lowpass channel, just make appropriate substitutions).

## 7.2.2 Complementary Channels and Their Properties

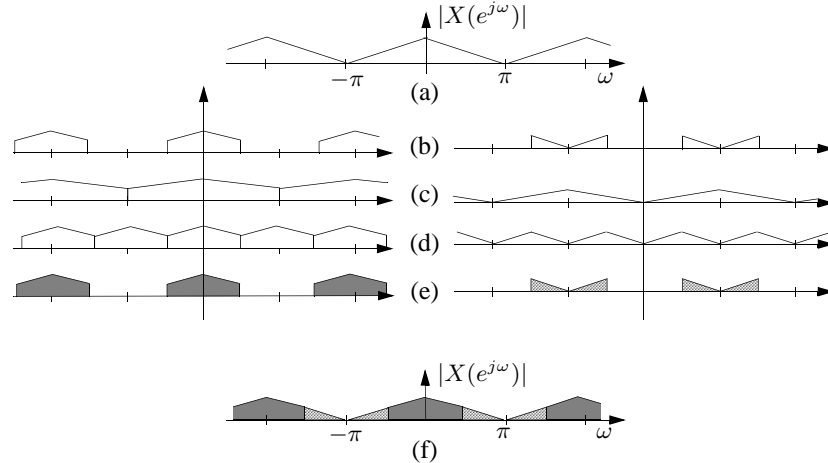
While we have discussed which properties each channel has to satisfy on its own, we now discuss what they have to satisfy with respect to each other to build an orthonormal basis. Intuitively, one channel has to keep what the other throws away; in other words, that channel should project to a subspace orthogonal to the range of the projection operator of the other. For example, given  $P_V$ ,  $P_W$  should project onto the leftover space between  $\ell^2(\mathbb{Z})$  and  $P_V \ell^2(\mathbb{Z})$ .

Thus, we start by assuming our filter bank in Figure 7.1(a) implements an orthonormal basis, which means that the set of basis sequences  $\{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}$  is an orthonormal set, compactly represented by (7.8). We have already used the orthonormality of the set  $\{g_{n-2k}\}_{k \in \mathbb{Z}}$  in (7.13) as well as the orthonormality of the set  $\{h_{n-2k}\}_{k \in \mathbb{Z}}$  in (7.14). What is left is that these two sets are orthogonal to each other, the third equation in (7.8).

**Orthogonality of the Lowpass and Highpass Filters** Using similar methods as before, we summarize the lowpass and highpass sequences must satisfy:

$$\begin{array}{ccc} \text{Matrix View} & & D_2 H^T G U_2 = 0 \\ \downarrow \text{ZT} & & G(z)H(z^{-1}) + G(-z)H(-z^{-1}) = 0 \\ \text{DTFT} & & G(e^{j\omega})H(e^{-j\omega}) + G(e^{j(\omega+\pi)})H(e^{-j(\omega+\pi)}) = 0 \\ \uparrow & & \end{array} \quad (7.22)$$





**Figure 7.5:** Two-channel decomposition of a sequence using ideal filters. Left side depicts the process in the lowpass channel, while the right side depicts the process in the highpass channel. (a) Original spectrum. (b) Spectra after filtering. (c) Spectra after downsampling. (d) Spectra after upsampling. (e) Spectra after interpolation filtering. (f) Reconstructed spectrum.

**Deterministic Crosscorrelation of the Lowpass and Highpass Filters** Instead of the deterministic autocorrelation properties of an orthogonal filter, we look at the deterministic crosscorrelation properties of two filters orthogonal to each other:

$$\begin{array}{ccc}
 \langle g_n, h_{n-2k} \rangle = c_{2k} = 0 & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_2 C U_2 = 0 \\ C(z) + C(-z) = 0 \\ C(e^{j\omega}) + C(e^{j(\omega+\pi)}) = 0 \end{array} \quad (7.23)
 \end{array}$$

In the above,  $C = H^T G$  is the deterministic crosscorrelation operator, and the deterministic crosscorrelation is given by (2.99). In particular, all the even terms of  $c$  are equal to zero.

### 7.2.3 Orthogonal Two-Channel Filter Bank

We are now ready to put together everything we have developed so far. We have shown that the sequences  $\{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}$  form an orthonormal set. What is left to show is completeness: any sequence from  $\ell^2(\mathbb{Z})$  can be represented using the orthonormal basis built by our orthogonal two-channel filter bank. To do this, we must be more specific, that is, we must have an explicit form of the filters involved.

In essence, we start with an educated guess (and it will turn out to be unique, Theorem 7.2), inspired by what we have seen in the Haar case. We can also strengthen our intuition by considering a two-channel filter bank with ideal filters

as in Figure 7.5. If we are given an orthogonal lowpass filter  $g$ , can we say anything about an appropriate orthogonal highpass filter  $h$  such that the two together build an orthonormal basis? A good approach would be to shift the spectrum of the lowpass filter by  $\pi$ , leading to the highpass filter. In time domain, this is equivalent to multiplying  $g_n$  by  $(-1)^n$ . Because of the orthogonality of the lowpass and highpass filters, we also reverse the impulse response of  $g$ . We will then need to shift the filter to make it causal again. Based on this discussion, we now show how, given an orthogonal filter  $g$ , it completely specifies an orthogonal two-channel filter bank implementing an orthonormal basis for  $\ell^2(\mathbb{Z})$ :

**THEOREM 7.1 (ORTHOGONAL TWO-CHANNEL FILTER BANK)** Given is an FIR filter  $g$  of even length  $L = 2\ell$ ,  $\ell \in \mathbb{Z}^+$ , orthonormal to its even shifts as in (7.13). Choose

$$h_n = \pm(-1)^n g_{-n+L-1} \quad \xleftrightarrow{ZT} \quad H(z) = \mp z^{-L+1} G(-z^{-1}). \quad (7.24)$$

Then,  $\{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}$  is an orthonormal basis for  $\ell^2(\mathbb{Z})$ , implemented by an orthogonal filter bank specified by analysis filters  $\{g_{-n}, h_{-n}\}$  and synthesis filters  $\{g_n, h_n\}$ . The expansion splits  $\ell^2(\mathbb{Z})$  as

$$\ell^2(\mathbb{Z}) = V \oplus W, \quad \text{with} \quad \begin{aligned} V &= \overline{\text{span}}(\{g_{n-2k}\}_{k \in \mathbb{Z}}), \\ W &= \overline{\text{span}}(\{h_{n-2k}\}_{k \in \mathbb{Z}}). \end{aligned} \quad (7.25)$$

*Proof.* To prove the theorem, we must prove that (i)  $\{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}$  is an orthonormal set and (ii) it is complete. The sign  $\pm$  in (7.24) just changes phase; assuming  $G(1) = \sqrt{2}$ , if the sign is positive,  $H(1) = \sqrt{2}$ , and if the sign is negative,  $H(1) = -\sqrt{2}$ . Most of the time we will implicitly assume the sign to be positive; the proof of the theorem does not change in either case.

- (i) To prove that  $\{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}$  is an orthonormal set, we must prove (7.8). The first condition is satisfied by assumption. To prove the second, that is,  $h$  is orthogonal to its even shifts, we must prove one of the conditions in (7.14). The definition of  $h$  in (7.24) implies

$$H(z)H(z^{-1}) = G(-z)G(-z^{-1}), \quad (7.26)$$

and thus,

$$H(z)H(z^{-1}) + H(-z^{-1})H(-z^{-1}) = G(-z)G(-z^{-1}) + G(z)G(z^{-1}) \stackrel{(a)}{=} 2,$$

where (a) follows from (7.13).

To prove the third condition in (7.8), that is,  $h$  is orthogonal to  $g$  and all its even shifts, we must prove one of the conditions in (7.22):

$$\begin{aligned} G(z)H(z^{-1}) + G(-z)H(-z^{-1}) &\stackrel{(a)}{=} -z^{L-1}G(z)G(-z) + (-1)^L z^{L-1}G(-z)G(z) \\ &= -z^{L-1}G(z)G(-z) + z^{L-1}G(z)G(-z) \stackrel{(b)}{=} 0, \end{aligned}$$

where (a) follows from (7.24); and (b) from  $L = 2\ell$  even.

## 7.2. Orthogonal Two-Channel Filter Banks

579

- (ii) To prove completeness, we prove that perfect reconstruction holds for any  $x \in \ell^2(\mathbb{Z})$  (an alternative would be to prove Parseval's equality  $\|x\|^2 = \|x_V\|^2 + \|x_W\|^2$ ). What we do is find  $z$ -domain expressions for  $X_V(z)$  and  $X_W(z)$  and prove they sum up to  $X(z)$ . We start with the lowpass branch. In the lowpass channel, the input  $X(z)$  is filtered by  $G(z^{-1})$ , and is then down- and upsampled, followed by filtering with  $G(z)$  (and similarly for the highpass channel). Thus, the  $z$ -transforms of  $x_V$  and  $x_W$  are:

$$X_V(z) = \frac{1}{2}G(z) [G(z^{-1})X(z) + G(-z^{-1})X(-z)], \quad (7.27a)$$

$$X_W(z) = \frac{1}{2}H(z) [H(z^{-1})X(z) + H(-z^{-1})X(-z)]. \quad (7.27b)$$

The output of the filter bank is the sum of  $x_V$  and  $x_W$ :

$$\begin{aligned} X_V(z) + X_W(z) &= \frac{1}{2} \underbrace{[G(z)G(-z^{-1}) + H(z)H(-z^{-1})]}_{S(z)} X(-z) \\ &\quad + \frac{1}{2} \underbrace{[G(z)G(z^{-1}) + H(z)H(z^{-1})]}_{T(z)} X(z). \end{aligned} \quad (7.28)$$

Substituting (7.24) into the above equation, we get:

$$\begin{aligned} S(z) &= G(z)G(-z^{-1}) + H(z)H(-z^{-1}) \\ &\stackrel{(a)}{=} G(z)G(-z^{-1}) + [-z^{-L+1}G(-z^{-1})] [-(-z^{-1})^{-L+1}G(z)] \\ &= [1 + (-1)^{-L+1}] G(z)G(-z^{-1}) \stackrel{(b)}{=} 0, \end{aligned} \quad (7.29a)$$

$$\begin{aligned} T(z) &= G(z)G(z^{-1}) + H(z)H(z^{-1}) \\ &\stackrel{(c)}{=} G(z)G(z^{-1}) + G(-z^{-1})G(-z) \stackrel{(d)}{=} 2, \end{aligned} \quad (7.29b)$$

where (a) follows from (7.24); (b) from  $L = 2\ell$  is even; (c) from (7.26); and (d) from (7.13). Substituting this back into (7.28), we get

$$X_V(z) + X_W(z) = X(z), \quad (7.30)$$

proving perfect reconstruction, or, in other words, the assertion in the theorem statement that the expansion can be implemented by an orthogonal filter bank.

To show (7.25), we write (7.30) in the original domain as in (7.11):

$$x_n = \underbrace{\sum_{k \in \mathbb{Z}} \alpha_k g_{n-2k}}_{x_{V,n}} + \underbrace{\sum_{k \in \mathbb{Z}} \beta_k h_{n-2k}}_{x_{W,n}}, \quad (7.31)$$

showing that any sequence  $x \in \ell^2(\mathbb{Z})$  can be written as a sum of its projections onto two subspaces  $V$  and  $W$ , and these subspaces add up to  $\ell^2(\mathbb{Z})$ .  $V$  and  $W$  are orthogonal from (7.22) proving (7.25).

In the theorem,  $L$  is an even integer, which is a requirement for FIR filters of lengths greater than 1 (see Exercise 7.2). Moreover, the choice (7.24) is unique; this will be shown in Theorem 7.2. Table 7.9 summarizes various properties of orthogonal, two-channel filter banks we covered until now.

Along with the time reversal and shift, the other qualitative feature of (7.24) is modulation by  $e^{jn\pi} = (-1)^n$  (mapping  $z \rightarrow -z$  in the  $z$  domain, see (2.136)). As we said, this makes  $h$  a highpass filter when  $g$  is a lowpass filter. As an example, if we apply Theorem 7.1 to the Haar lowpass filter from (7.1a), we obtain the Haar highpass filter from (7.1b).

In applications, filters are causal. To implement a filter bank with causal filters, we make analysis filters causal (we already assumed the synthesis ones are) by shifting them both by  $(-L + 1)$ . Beware that such an implementation implies perfect reconstruction within a shift (delay), and the orthonormal basis expansion is not technically valid anymore. However, in applications this is often done, as the output sequence is a perfect replica of the input one, within a shift:  $\hat{x}_n = x_{n-L+1}$ .

### 7.2.4 Polyphase View of Orthogonal Filter Banks

As we saw in Section 2.7, downsampling introduces periodic shift variance into the system. To deal with this, we often analyze multirate systems in polyphase domain, as discussed in Section 2.7.6. The net result is that the analysis of a single-input, single-output, periodically shift-varying system is equivalent to the analysis of a multiple-input, multiple-output, shift-invariant system.

**Polyphase Representation of an Input Sequence** For two-channel filter banks, a polyphase decomposition of the input sequence is achieved by simply splitting it into its even- and odd-indexed subsequences as in (2.210), the main idea being that the sequence can be recovered from the two subsequences by upsampling, shifting and summing up, as we have seen in Figure 2.23. This simple process is called a polyphase transform (forward and inverse).

**Polyphase Representation of a Synthesis Filter Bank** To define the polyphase decomposition of the synthesis filters, we use the expressions for upsampling followed by filtering from (2.215):

$$g_{0,n} = g_{2n} \xleftrightarrow{\text{ZT}} G_0(z) = \sum_{n \in \mathbb{Z}} g_{2n} z^{-n}, \quad (7.32a)$$

$$g_{1,n} = g_{2n+1} \xleftrightarrow{\text{ZT}} G_1(z) = \sum_{n \in \mathbb{Z}} g_{2n+1} z^{-n}, \quad (7.32b)$$

$$G(z) = G_0(z^2) + z^{-1}G_1(z^2), \quad (7.32c)$$

where we split each synthesis filter into its even and odd subsequence as we have done for the input sequence  $x$ . Analogous relations hold for the highpass filter  $h$ . We can now define a *polyphase matrix*  $\Phi_p(z)$ :

$$\Phi_p(z) = \begin{bmatrix} G_0(z) & H_0(z) \\ G_1(z) & H_1(z) \end{bmatrix}. \quad (7.33)$$

As we will see in (7.37), such a matrix allows for a compact representation, analysis and computing projections in the polyphase domain.

**Polyphase Representation of an Analysis Filter Bank** The matrix in (7.33) is on the synthesis side; to get it on the analysis side, we can use the fact that this is an orthogonal filter bank. Thus, we can write

$$\tilde{G}(z) = G(z^{-1}) = G_0(z^{-2}) + zG_1(z^{-2}).$$

In other words, the polyphase components of the analysis filter are, not surprisingly, time-reversed versions of the polyphase components of the synthesis filter. We can summarize this as (we could have obtained the same result using the expression for polyphase representation of filtering followed by downsampling (2.221)):

$$\tilde{g}_{0,n} = \tilde{g}_{2n} = g_{-2n} \xleftrightarrow{\text{ZT}} \tilde{G}_0(z) = \sum_{n \in \mathbb{Z}} g_{-2n} z^{-n}, \quad (7.34a)$$

$$\tilde{g}_{1,n} = \tilde{g}_{2n-1} = g_{-2n+1} \xleftrightarrow{\text{ZT}} \tilde{G}_1(z) = \sum_{n \in \mathbb{Z}} g_{-2n+1} z^{-n}, \quad (7.34b)$$

$$\tilde{G}(z) = G_0(z^{-2}) + zG_1(z^{-2}), \quad (7.34c)$$

with analogous relations for the highpass filter  $\tilde{h}$ , yielding the expression for the analysis polyphase matrix

$$\tilde{\Phi}_p(z) = \begin{bmatrix} \tilde{G}_0(z) & \tilde{H}_0(z) \\ \tilde{G}_1(z) & \tilde{H}_1(z) \end{bmatrix} = \begin{bmatrix} G_0(z^{-1}) & H_0(z^{-1}) \\ G_1(z^{-1}) & H_1(z^{-1}) \end{bmatrix} = \Phi_p(z^{-1}). \quad (7.35)$$

A block diagram of the polyphase implementation of the system is given in Figure 7.6. The left part shows the reconstruction of the original sequence using the synthesis polyphase matrix.<sup>107</sup> The right part shows the computation of expansion coefficient sequences  $\alpha$  and  $\beta$ ; note that as usual, the analysis matrix (polyphase in this case) is taken as a transpose, as it operates on the input sequence. To check that, compute these expansion coefficient sequences:

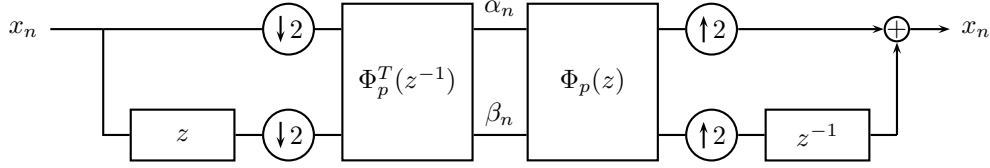
$$\begin{aligned} \begin{bmatrix} \alpha(z) \\ \beta(z) \end{bmatrix} &= \Phi_p^T(z^{-1}) \begin{bmatrix} X_0(z) \\ X_1(z) \end{bmatrix} = \begin{bmatrix} G_0(z^{-1}) & G_1(z^{-1}) \\ H_0(z^{-1}) & H_1(z^{-1}) \end{bmatrix} \begin{bmatrix} X_0(z) \\ X_1(z) \end{bmatrix} \\ &= \begin{bmatrix} G_0(z^{-1})X_0(z) + G_1(z^{-1})X_1(z) \\ H_0(z^{-1})X_0(z) + H_1(z^{-1})X_1(z) \end{bmatrix}. \end{aligned} \quad (7.36)$$

We can obtain exactly the same expressions if we substitute (7.34) into the expression for filtering followed by downsampling by 2 in (2.196a).

**Polyphase Representation of an Orthogonal Filter Bank** The above polyphase expressions allow us now to compactly represent an orthogonal two-channel filter bank in the polyphase domain:

$$X(z) = \begin{bmatrix} 1 & z^{-1} \end{bmatrix} \Phi_p(z^2) \Phi_p^T(z^{-2}) \begin{bmatrix} X_0(z^2) \\ X_1(z^2) \end{bmatrix}. \quad (7.37)$$

<sup>107</sup>A comment is in order: we typically put the lowpass filter in the lower branch, but in matrices it appears in the first row/column, leading to a slight inconsistency when the filter bank is depicted in the polyphase domain.



**Figure 7.6:** Polyphase representation of a two-channel orthogonal filter bank.

From (7.24), we get that the polyphase components of  $H$  are

$$H_0(z) = \pm z^{-L/2+1} G_1(z^{-1}), \quad (7.38a)$$

$$H_1(z) = \mp z^{-L/2+1} G_0(z^{-1}), \quad (7.38b)$$

leading to the polyphase matrix

$$\Phi_p(z) = \begin{bmatrix} G_0(z) & \pm z^{-L/2+1} G_1(z^{-1}) \\ G_1(z) & \mp z^{-L/2+1} G_0(z^{-1}) \end{bmatrix}. \quad (7.39)$$

Since  $g$  is orthogonal to its even translates, substitute (7.32) into the  $z$ -domain version of (7.13) to get the condition for orthogonality of a filter in polyphase form:

$$G_0(z)G_0(z^{-1}) + G_1(z)G_1(z^{-1}) = 1. \quad (7.40)$$

Using this, the determinant of  $\Phi_p(z)$  becomes  $-z^{-L/2+1}$ . From (7.37), the polyphase matrix  $\Phi_p(z)$  satisfies the following:

$$\Phi_p(z)\Phi_p^T(z^{-1}) = I, \quad (7.41)$$

a paraunitary matrix as in (2.294a). In fact, (7.39), together with (7.40), define the most general  $2 \times 2$ , real-coefficient, causal FIR lossless matrix, a fact we summarize in form of a theorem, the proof of which can be found in [162]:

**THEOREM 7.2 (GENERAL FORM OF A PARAUNITARY MATRIX)** The most general  $2 \times 2$ , real-coefficient, causal FIR lossless matrix is given by (7.39), where  $G_0$  and  $G_1$  satisfy (7.40) and  $L/2 - 1$  is the order of  $G_0(z)$ ,  $G_1(z)$ .

**EXAMPLE 7.1 (HAAR FILTER BANK IN POLYPHASE FORM)** The Haar filters (7.1) are extremely simple in polyphase form: Since they are both of length 2, their polyphase components are of length 1. The polyphase matrix is simply

$$\Phi_p(z) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \quad (7.42)$$

The form of the polyphase matrix for the Haar orthonormal basis is exactly the same as the Haar orthonormal basis for  $\mathbb{R}^2$ , or one block of the Haar orthonormal

## 7.2. Orthogonal Two-Channel Filter Banks

583

basis infinite matrix  $\Phi$  from (I.7). This is true only when a filter bank implements the so-called *block transform*, that is, when the nonzero support of the basis sequences is equal to the sampling factor, 2 in this case.

The polyphase notation and the associated matrices are powerful tools to derive filter bank results. We now rephrase what it means for a filter bank to be orthogonal—implement an orthonormal basis, in polyphase terms.

**THEOREM 7.3 (PARAUNITARY POLYPHASE MATRIX AND ORTHONORMAL BASIS)**

A  $2 \times 2$  polyphase matrix  $\Phi_p(z)$  is paraunitary if and only if the associated two-channel filter bank implements an orthonormal basis for  $\ell^2(\mathbb{Z})$ .

*Proof.* If the polyphase matrix is paraunitary, then the expansion it implements is complete, due to (7.41). To prove that the expansion is an orthonormal basis, we must show that the basis sequences form an orthonormal set. From (7.39) and (7.41), we get (7.40). Substituting this into the  $z$ -domain version of (7.13), we see that it holds, and thus  $g$  and its even shifts form an orthonormal set. Because  $h$  is given in terms of  $g$  as (7.24),  $h$  and its even shifts form an orthonormal set as well. Finally, because of the way  $h$  is defined,  $g$  and  $h$  are orthogonal by definition and so are their even shifts.

The argument in the other direction is similar; we start with an orthonormal basis implemented by a two-channel filter bank. That means we have template sequences  $g$  and  $h$  related via (7.24), and their even shifts, all together forming an orthonormal basis. We can now translate those conditions into  $z$ -transform domain using (7.13) and derive the corresponding polyphase-domain versions, such as the one in (7.40). These lead to the polyphase matrix being paraunitary.

We have seen in Chapter 2 that we can characterize vector sequences using deterministic autocorrelation matrices (see Table 2.13). We use this now to describe the deterministic autocorrelation of a vector sequence of expansion coefficients  $[\alpha_n \ \beta_n]^T$ , as

$$\begin{aligned}
 A_{p,\alpha}(z) &= \begin{bmatrix} A_\alpha(z) & C_{\alpha,\beta}(z) \\ C_{\beta,\alpha}(z) & A_\beta(z) \end{bmatrix} = \begin{bmatrix} \alpha(z) \alpha(z^{-1}) & \alpha(z) \beta(z^{-1}) \\ \beta(z) \alpha(z^{-1}) & \beta(z) \beta(z^{-1}) \end{bmatrix} \\
 &= \begin{bmatrix} \alpha(z) \\ \beta(z) \end{bmatrix} \begin{bmatrix} \alpha(z^{-1}) & \beta(z^{-1}) \end{bmatrix} \\
 &\stackrel{(a)}{=} \Phi_p^T(z^{-1}) \begin{bmatrix} X_0(z) \\ X_1(z) \end{bmatrix} \begin{bmatrix} X_1(z^{-1}) & X_0(z^{-1}) \end{bmatrix} \Phi_p(z) \\
 &= \Phi_p^T(z^{-1}) A_{p,x}(z) \Phi_p(z),
 \end{aligned} \tag{7.43}$$

where (a) follows from (7.36), and  $A_{p,x}$  is the deterministic autocorrelation matrix of the vector of polyphase components of  $x$ . This deterministic autocorrelation matrix can be seen as a *filtered* deterministic autocorrelation of the input. We now have the following result:

**THEOREM 7.4 (FILTERED DETERMINISTIC AUTOCORRELATION MATRIX)** Given is a  $2 \times 2$  paraunitary polyphase matrix  $\Phi_p(e^{j\omega})$ . Then the filtered deterministic autocorrelation matrix,  $A_{p,\alpha}(e^{j\omega})$ , is positive semidefinite.

*Proof.* Since  $\Phi_p(z)$  is paraunitary,  $\Phi_p(e^{j\omega})$  is unitary on the unit circle. This further means that:

$$\begin{bmatrix} \cos \theta & \sin \theta \end{bmatrix} \Phi_p^T(e^{-j\omega}) = \begin{bmatrix} \cos \phi & \sin \phi \end{bmatrix}, \quad (7.44)$$

for some  $\phi$ . We can now write:

$$\begin{aligned} \begin{bmatrix} \cos \theta & \sin \theta \end{bmatrix} A_{p,\alpha}(e^{j\omega}) \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} &\stackrel{(a)}{=} \begin{bmatrix} \cos \theta & \sin \theta \end{bmatrix} \Phi_p^T(e^{-j\omega}) A_{p,x}(e^{j\omega}) \Phi_p(e^{j\omega}) \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} \\ &\stackrel{(b)}{=} \begin{bmatrix} \cos \phi & \sin \phi \end{bmatrix} A_{p,x}(e^{j\omega}) \begin{bmatrix} \cos \phi \\ \sin \phi \end{bmatrix} \stackrel{(c)}{\geq} 0, \end{aligned}$$

where (a) follows from (7.43); (b) from (7.44); and (c) from TBD, proving the theorem.

### 7.2.5 Polynomial Approximation by Filter Banks

An important class of orthogonal filter banks are those that have polynomial approximation properties; these filter banks will approximate polynomials of a certain degree<sup>108</sup> in the lowpass (coarse) branch, while, at the same time, blocking those same polynomials in the highpass (detail) branch. To derive these filter banks, we recall what we have learned in Section 2.B.1: Convolution of a polynomial sequence  $x$  with a differencing filter  $(\delta_n - \delta_{n-1})$ , or, multiplication of  $X(z)$  by  $(1 - z^{-1})$ , reduces the degree of the polynomial by 1. In general, to block a polynomial of degree  $(N - 1)$ ,  $x_n = \sum_{k=0}^{N-1} a_k n^k$ , we need a filter of the form:

$$(1 - z^{-1})^N R'(z). \quad (7.45)$$

Let us now apply what we just learned to two-channel orthogonal filter banks with polynomial sequences as inputs. We will construct the analysis filter in the highpass branch to have  $N$  zeros at  $z = 1$ , thus blocking polynomials of degree up to  $(N - 1)$ . Of course, since the filter bank is perfect reconstruction, whatever disappeared in the highpass branch must be preserved in the lowpass one; thus, the lowpass branch will reconstruct polynomials of degree  $(N - 1)$ . In other words,  $x_V$  will be a polynomial approximation of the input sequence a certain degree.

To construct such a filter bank, we start with the analysis highpass filter  $\tilde{h}$  which must be of the form (7.45); we write it as:

$$\tilde{H}(z) \stackrel{(a)}{=} (1 - z^{-1})^N \underbrace{\mp z^{L-1} R(-z)}_{R'(z)} = \mp z^{L-1} (1 - z^{-1})^N R(-z) \stackrel{(b)}{=} \mp z^{L-1} G(-z),$$

<sup>108</sup>We restrict our attention to finitely-supported polynomial sequences, ignoring the boundary issues. If this were not the case, these sequences would not belong to any  $\ell^p$  space.



## 7.3. Design of Orthogonal Two-Channel Filter Banks

585

where in (a) we have chosen  $R'(z)$  to lead to a simple form of  $G(z)$  in what follows; and (b) follows from Table 7.9, allowing us to directly read the synthesis lowpass as

$$G(z) = (1 + z^{-1})^N R(z). \quad (7.46)$$

If we maintain the convention that  $g$  is causal and of length  $L$ , then  $R(z)$  is a polynomial in  $z^{-1}$  of degree  $(L - 1 - N)$ . Of course,  $R(z)$  has to be chosen appropriately, so as to obtain an orthogonal filter bank.

Putting at least one zero at  $z = -1$  in  $G(z)$  makes a lot of signal processing sense. After all,  $z = -1$  corresponds to  $\omega = \pi$ , the maximum discrete frequency; it is thus natural for a lowpass filter to have a zero at  $z = -1$  and block that highest frequency. Putting more than one zero at  $z = -1$  has further approximation advantages, as the Proposition 7.5 specifies, and as we will see in wavelet constructions in later chapters.

**THEOREM 7.5 (POLYNOMIAL REPRODUCTION)** Given is an orthogonal filter bank in which the synthesis lowpass filter  $G(z)$  has  $N$  zeros at  $z = -1$ . Then polynomial sequences up to degree  $(N - 1)$  and of finite support are reproduced in the lowpass approximation subspace spanned by  $\{g_{n-2k}\}_{k \in \mathbb{Z}}$ .

*Proof.* By assumption, the synthesis filter  $G(z)$  is given by (7.46). From Table 7.9, the analysis highpass filter is of the form  $\mp z^{L-1} G(-z)$ , which means it has a factor  $(1 - z^{-1})^N$ , that is, it has  $N$  zeros at  $z = 1$ . From our discussion, this factor annihilates a polynomial input of degree  $(N - 1)$ , resulting in  $\beta = 0$  and  $x_W = 0$ . Because of the perfect reconstruction property,  $x = x_V$ , showing that the polynomial sequences are reproduced by a linear combination of  $\{g_{n-2k}\}_{k \in \mathbb{Z}}$ , as in (7.11a).

Polynomial reproduction by the lowpass channel and polynomial cancellation in the highpass channel are basic features in wavelet approximations. In particular, the cancellation of polynomials of degree  $(N - 1)$  is also called the *zero-moment property* of the filter (see (2.139a)):

$$m_k = \sum_{n \in \mathbb{Z}} n^k h_n = 0, \quad k = 0, 1, \dots, N - 1, \quad (7.47)$$

that is,  $k$ th-order moments of  $h$  up to  $(N - 1)$  are zero (see Exercise 7.6).

## 7.3 Design of Orthogonal Two-Channel Filter Banks

To design a two-channel orthogonal filter bank, it suffices to design one orthogonal filter—the lowpass synthesis  $g$  with the  $z$ -transform  $G(z)$  satisfying (7.13); we have seen how the other three filters follow (Table 7.9). The design is based on (1) finding a deterministic autocorrelation function satisfying (7.15) (it is symmetric, positive semi-definite and has a single nonzero even-indexed coefficient; and (2) factoring that deterministic autocorrelation  $A(z) = G(z)G(z^{-1})$  into its spectral factors (many factorizations are possible, see Section 2.5).<sup>109</sup>

<sup>109</sup>Recall that we only consider real-coefficient filters, thus  $a$  is symmetric and not Hermitian symmetric.

We consider three different designs. The first tries to approach an ideal half-band lowpass filter, the second aims at polynomial approximation, while the third uses lattice factorization in polyphase domain.

### 7.3.1 Lowpass Approximation Design

Assume we wish to get our lowpass synthesis filter  $G(e^{j\omega})$  to be as close as possible to an ideal lowpass halfband filter as in TBD. Since according to (2.96) the deterministic autocorrelation of  $g$  can be expressed in the DTFT domain as  $A(e^{j\omega}) = |G(e^{j\omega})|^2$ , this deterministic autocorrelation is an ideal lowpass halfband function as well:

$$A(e^{j\omega}) = \begin{cases} 2, & \text{if } |\omega| < \pi/2; \\ 0, & \text{otherwise.} \end{cases} \quad (7.48)$$

From Table 2.5, the deterministic autocorrelation sequence is

$$a_n = \frac{1}{2\pi} \int_{-\pi/2}^{\pi/2} 2 e^{jn\omega} d\omega = \text{sinc}(n\pi/2), \quad (7.49)$$

a valid deterministic autocorrelation; it has a single nonzero even-indexed coefficient ( $a_0 = 1$ ) and is positive semi-definite. To get a realizable function, we apply a symmetric window function  $w$  that decays to zero. The new deterministic autocorrelation  $a'$  is the pointwise product

$$a'_n = a_n w_n. \quad (7.50)$$

Clearly,  $a'$  is symmetric and still has a single nonzero even-indexed coefficient. However, this is not enough for  $a'$  to be a deterministic autocorrelation. We can see this in frequency domain,

$$A'(e^{j\omega}) \stackrel{(a)}{=} \frac{1}{2\pi} A(e^{j\omega}) * W(e^{j\omega}), \quad (7.51)$$

where we used the convolution in frequency property (2.94). In general, (7.51) is not nonnegative for all frequencies anymore, and thus not a valid deterministic autocorrelation. One easy way to enforce nonnegativity is to choose  $W(e^{j\omega})$  itself positive, for example as the deterministic autocorrelation of another window  $w'$ , or

$$W(e^{j\omega}) = |W'(e^{j\omega})|^2.$$

If  $w'$  is of norm 1, then  $w_0 = 1$ , and from (7.50),  $a'_0 = 1$  as well. Therefore, since  $A(e^{j\omega})$  is real and positive,  $A'(e^{j\omega})$  will be as well. The resulting sequence  $a'$  and its  $z$ -transform  $A'(z)$  can then be used in spectral factorization (see Section 2.5.3) to obtain an orthogonal filter  $g$ .

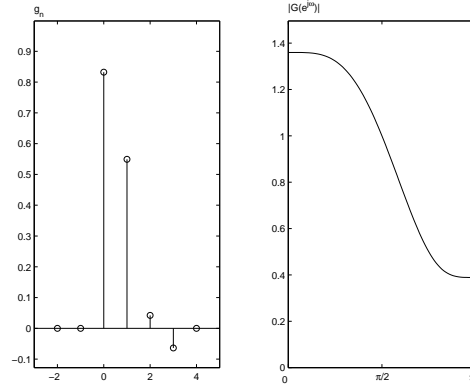
#### EXAMPLE 7.2 (LOWPASS APPROXIMATION DESIGN OF ORTHOGONAL FILTERS)

We design a length-4 filter by the lowpass approximation method. Its deterministic autocorrelation is of length 7 with the target impulse response obtained by evaluating (7.49):

$$a = \left[ \dots \quad 0 \quad -\frac{2}{3\pi} \quad 0 \quad \frac{2}{\pi} \quad \boxed{1} \quad \frac{2}{\pi} \quad 0 \quad -\frac{2}{3\pi} \quad 0 \quad \dots \right]^T.$$

## 7.3. Design of Orthogonal Two-Channel Filter Banks

587



**Figure 7.7:** Orthogonal filter design based on lowpass approximation in Example 7.2. (a) Impulse response. (b) Frequency response.

For the window  $w$ , we take it to be the deterministic autocorrelation of the sequence  $w'_n$ , which is specified by  $w'_n = 1/2$  for  $0 \leq n \leq 3$ , and  $w'_n = 0$  otherwise:

$$w = \left[ \dots \quad 0 \quad 0 \quad \frac{1}{4} \quad \frac{1}{2} \quad \frac{3}{4} \quad \boxed{1} \quad \frac{3}{4} \quad \frac{1}{2} \quad \frac{1}{4} \quad 0 \quad 0 \quad \dots \right]^T.$$

Using (7.50), we obtain the new deterministic autocorrelation of the lowpass filter as

$$a' = \left[ \dots \quad 0 \quad -\frac{1}{6\pi} \quad 0 \quad \frac{3}{2\pi} \quad \boxed{1} \quad \frac{3}{2\pi} \quad 0 \quad -\frac{1}{6\pi} \quad 0 \quad \dots \right]^T.$$

Factoring this deterministic autocorrelation (requires numerical polynomial root finding) gives

$$g \approx \left[ \dots \quad 0 \quad \boxed{0.832} \quad 0.549 \quad 0.0421 \quad -0.0637 \quad 0 \quad \dots \right]^T.$$

The impulse response and frequency response of  $g$  are shown in Figure 7.7.

The method presented is very simple, and does not lead to the best designs. For better designs, one uses standard filter design procedures followed by adjustments to ensure positivity. For example, consider (7.51) again, and define

$$\min_{\omega \in [-\pi, \pi]} A'(e^{j\omega}) = \varepsilon.$$

If  $\varepsilon \geq 0$ , we are done, otherwise, we simply choose a new function

$$A''(e^{j\omega}) = A'(e^{j\omega}) - \varepsilon,$$

which is now nonnegative, allowing us to perform spectral factorization. Filters designed using this method are tabulated in [160].

### 7.3.2 Polynomial Approximation Design

Recall that a lowpass filter  $G(z)$  with  $N$  zeros at  $z = -1$  as in (7.46) reproduces polynomials up to degree  $(N-1)$ . Thus, the goal of this design procedure is to find a deterministic autocorrelation  $A(z)$  of the form

$$A(z) = G(z)G(z^{-1}) = (1 + z^{-1})^N(1 + z)^N Q(z), \quad (7.52)$$

with  $Q(z)$  chosen such that (7.15) is satisfied, that is,

$$A(z) + A(-z) = 2, \quad (7.53)$$

$Q(z) = Q(z^{-1})$  ( $q_n$  symmetric in time domain), and  $Q(z)$  is nonnegative on the unit circle. Satisfying these conditions allows one to find a spectral factor of  $A(z)$  with  $N$  zeros at  $z = -1$ , and this spectral factor is the desired orthogonal filter. We illustrate this procedure through an example.

**EXAMPLE 7.3 (POLYNOMIAL APPROXIMATION DESIGN OF ORTHOGONAL FILTERS)**

We will design a filter  $g$  such that it reproduces linear polynomials, that is,  $N = 2$ :

$$A(z) = (1 + z^{-1})^2(1 + z)^2 Q(z) = (z^{-2} + 4z^{-1} + 6 + 4z + z^2) Q(z).$$

Can we now find  $Q(z)$  so as to satisfy (7.53), in particular, a minimum-degree solution? We try with (remember  $q_n$  is symmetric)

$$Q(z) = az + b + az^{-1}$$

and compute  $A(z)$  as

$$A(z) = a(z^3 + z^{-3}) + (4a + b)(z^2 + z^{-2}) + (7a + 4b)(z + z^{-1}) + (8a + 6b).$$

To satisfy (7.53),  $A(z)$  must have a single nonzero even-indexed coefficient. We thus need to solve the following pair of equations:

$$\begin{aligned} 4a + b &= 0, \\ 8a + 6b &= 1, \end{aligned}$$

yielding  $a = -1/16$  and  $b = 1/4$ . Thus, our candidate factor is

$$Q(z) = \frac{1}{4} \left( -\frac{1}{4}z^{-1} + 1 - \frac{1}{4}z \right).$$

It remains to check whether  $Q(e^{j\omega})$  is nonnegative:

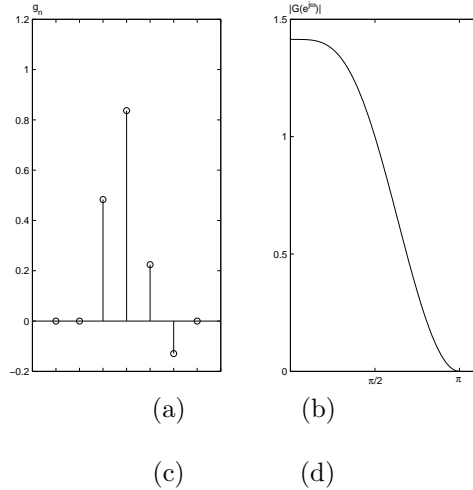
$$Q(e^{j\omega}) = \frac{1}{4} \left( 1 - \frac{1}{4}(e^{j\omega} + e^{-j\omega}) \right) = \frac{1}{4} \left( 1 - \frac{1}{2} \cos \omega \right) > 0$$

since  $|\cos(\omega)| \leq 1$ . So  $Q(z)$  is a valid deterministic autocorrelation and can be written as  $Q(z) = R(z)R(z^{-1})$ . Extracting its causal spectral factor

$$R(z) = \frac{1}{4\sqrt{2}}(1 + \sqrt{3} + (1 - \sqrt{3})z^{-1}),$$

## 7.3. Design of Orthogonal Two-Channel Filter Banks

589



**Figure 7.8:** Orthogonal filter design based on polynomial approximation in Example 7.3. (a) Impulse response. (b) Frequency response. (c) Linear  $x$  is preserved in  $V$ . (d) Only the linear portion of the quadratic  $x$  is preserved in  $V$ ; the rest shows in  $W$ .

the causal orthogonal lowpass filter with 2 zeros at  $z = -1$  becomes

$$\begin{aligned} G(z) &= (1 + z^{-1})^2 R(z) \\ &= \frac{1}{4\sqrt{2}} \left[ (1 + \sqrt{3}) + (3 + \sqrt{3})z^{-1} + (3 - \sqrt{3})z^{-2} + (1 - \sqrt{3})z^{-3} \right]. \end{aligned}$$

This filter is one of the filters from the Daubechies family of orthogonal filters. Its impulse and frequency responses are shown in Figure 7.8. The rest of the filters in the filter bank can be found from Table 7.9.

In the example, we saw that solving a linear system followed by spectral factorization were the key steps. In general, for  $G(z)$  with  $N$  zeros at  $z = -1$ , the minimum-degree  $R(z)$  to obtain an orthogonal filter is of degree  $(N - 1)$ , corresponding to  $N$  unknown coefficients.  $Q(z) = R(z)R(z^{-1})$  is obtained by solving an  $N \times N$  linear system (to satisfy  $A(z) + A(z) = 2$ ), followed by spectral factorization to produce the desired result. (It can be verified that  $Q(e^{j\omega}) \geq 0$ .) These steps are summarized in Table 7.2, while Table 7.3 gives filter-design examples.

Note that  $A(z)$  has the following form when evaluated on the unit circle:

$$A(e^{j\omega}) = 2^N (1 + \cos \omega)^N Q(e^{j\omega}),$$

with  $Q(e^{j\omega})$  real and positive. Since  $A(e^{j\omega})$  and its  $(2N - 1)$  derivatives are zero at  $\omega = \pi$ ,  $|G(e^{j\omega})|$  and its  $(N - 1)$  derivatives are zero at  $\omega = \pi$ . Moreover, because of the quadrature mirror formula (2.208),  $|G(e^{j\omega})|$  and its  $(N - 1)$  derivatives are zero at  $\omega = 0$  as well. These facts are the topic of Exercise 7.8.

Step	Operation
1.	Choose $N$ , the number of zeros at $z = -1$
2.	$G(z) = (1 + z^{-1})^N R(z)$ , where $R(z)$ is causal with powers $(0, -1, \dots, -N + 1)$
3.	$A(z) = (1 + z^{-1})^N (1 + z)^N Q(z)$ , where $Q(z)$ is symmetric and has powers $(-(N - 1), \dots, 0, \dots, (N + 1))$
4.	$A(z) + A(-z) = 2$ . This leads to $N$ linear constraints on the coefficients of $Q(z)$
5.	Solve the $N \times N$ linear system for the coefficients of $Q(z)$
6.	Take the spectral factor of $Q(z) = R(z)R(z^{-1})$ (for example, the minimum-phase factor, see Section 2.5)
7.	The minimum phase orthogonal filter is $G(z) = (1 + z^{-1})^N R(z)$

**Table 7.2:** Design of orthogonal lowpass filters with maximum number of zeros at  $z = -1$ .

	$L = 4$	$L = 6$	$L = 8$	$L = 10$	$L = 12$
$g_0$	0.482962913	0.332670553	0.230377813309	0.160102398	0.111540743350
$g_1$	0.836516304	0.806891509	0.714846570553	0.603829270	0.494623890398
$g_2$	0.224143868	0.459877502	0.630880767930	0.724308528	0.751133908021
$g_3$	-0.129409522	-0.135011020	-0.027983769417	0.138428146	0.315250351709
$g_4$		-0.085441274	-0.187034811719	-0.242294887	-0.226264693965
$g_5$		0.035226292	0.030841381836	-0.032244870	-0.129766867567
$g_6$			0.032883011667	0.077571494	0.097501605587
$g_7$			-0.010597401785	-0.006241490	0.027522865530
$g_8$				-0.012580752	-0.031582039318
$g_9$				0.003335725	0.000553842201
$g_{10}$					0.004777257511
$g_{11}$					-0.001077301085

**Table 7.3:** Orthogonal filters with maximum number of zeros at  $z = -1$  (from [39]). For a lowpass filter of even length  $L = 2\ell$ , there are  $L/2$  zeros at  $z = -1$ .

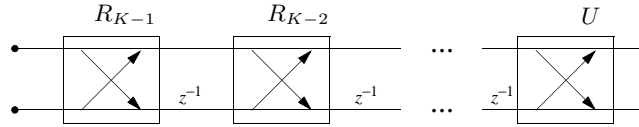
### 7.3.3 Lattice Factorization Design

When discussing the polyphase view of filter banks in Section 7.2.4, we saw that orthogonality of a two-channel filter bank is connected to its polyphase matrix being paraunitary. The following elegant factorization result is used in the design of that paraunitary matrix:

**THEOREM 7.6** The polyphase matrix of any real-coefficient, causal, FIR orthog-

## 7.3. Design of Orthogonal Two-Channel Filter Banks

591



**Figure 7.9:** Two-channel lattice factorization of paraunitary filter banks. The  $2 \times 2$  blocks  $R_k$  are rotation matrices, and  $U$  is a general unitary matrix (rotation or rotoinversion). The inputs are the polyphase components of the sequence  $x$ , and the output are the lowpass and highpass channels.

onal two-channel filter bank can be written as

$$\Phi_p(z) = U \prod_{k=1}^{K-1} \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} R_k, \quad (7.54)$$

where  $U$  is a general unitary matrix as in (1.218) (either a rotation as in (1.220a) or a rotoinversion (1.220b)), and  $R_k$ ,  $k = 1, 2, \dots, K-1$ , are rotation matrices as in (1.220a).

The resulting filters are of even length  $2K$  (see Exercise 7.2). That the above structure produces an orthogonal filter bank is clear as the corresponding polyphase matrix  $\Phi_p(z)$  is paraunitary. Proving that any orthogonal filter bank can be written in the form of (7.54) is a bit more involved. It is based on the result that for two, real-coefficient polynomials  $P_{K-1}$  and  $Q_{K-1}$  of degree  $(K-1)$ , with  $p_{K-1}(0) p_{K-1}(K-1) \neq 0$  (and  $P_{K-1}, Q_{K-1}$  are power complementary as in (2.208)), there exists another pair  $P_{K-2}, Q_{K-2}$  such that

$$\begin{bmatrix} P_{K-1}(z) \\ Q_{K-1}(z) \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} P_{K-2}(z) \\ z^{-1} Q_{K-2}(z) \end{bmatrix}. \quad (7.55)$$

Repeatedly applying the above result to (7.39) one obtains the lattice factorization given in (7.6). The details of the proof are given in [160].

Using the factored form, designing an orthogonal filter bank amounts to choosing  $U$  and a set of angles  $(\theta_0, \theta_1, \dots, \theta_{K-1})$ . For example, the Haar filter bank in lattice form amounts to keeping only the constant-matrix term,  $U$ , as in (7.42), a rotoinversion. The factored form also suggests a structure, called a lattice, convenient for hardware implementations (see Figure 7.9).

How do we impose particular properties, such as zeros at  $\omega = \pi$ , or,  $z = -1$ , for the lowpass filter  $G(z)$ ? Write the following set of equations:

$$G(z)|_{z=1} \stackrel{(a)}{=} (G_0(z^2) + z^{-1}G_1(z^2))|_{z=1} = G_0(1) + G_1(1) \stackrel{(b)}{=} \sqrt{2}, \quad (7.56a)$$

$$G(z)|_{z=-1} \stackrel{(c)}{=} (G_0(z^2) + z^{-1}G_1(z^2))|_{z=-1} = G_0(1) - G_1(1) \stackrel{(d)}{=} 0, \quad (7.56b)$$

where (a) and (c) follow from (7.32); (b) from (7.13) and the requirement that  $G(z)$  be 0 at  $z = -1$ , and similarly for (d). We can rewrite these compactly as:

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \underbrace{\begin{bmatrix} G_0(1) \\ G_0(1) \end{bmatrix}}_G = \begin{bmatrix} \sqrt{2} \\ 0 \end{bmatrix}. \quad (7.57)$$

The vector  $G$  above is just the first column of  $\Phi_p(z^2)|_{z=-1}$ , which, in turn, is either a product of (1)  $K$  rotations by  $(\theta_0, \theta_1, \dots, \theta_{K-1})$ , or, (2) one rotoinversion by  $\theta_0$  and  $K-1$  rotations  $(\theta_1, \dots, \theta_{K-1})$ . The solution to the above is:

$$\sum_{k=0}^{K-1} \theta_k = 2n\pi + \frac{\pi}{4}, \quad U \text{ is a rotation}, \quad (7.58a)$$

$$\theta_0 - \sum_{k=1}^{K-1} \theta_k = 2n\pi + \frac{\pi}{4}, \quad U \text{ is a rotoinversion}, \quad (7.58b)$$

for some  $n \in \mathbb{Z}$ . Imposing higher-order zeros at  $z = -1$ , as required for higher-order polynomial reproduction, leads to more complicated algebraic constraints. As an example, choosing  $\theta_0 = \pi/3$  and  $\theta_1 = -\pi/12$  leads to a double zero at  $z = -1$ , and is thus the lattice version of the filter designed in Example 7.3 (see Exercise 7.3). In general, design problems in lattice factored form are nonlinear and thus nontrivial.

## 7.4 Biorthogonal Two-Channel Filter Banks

While orthogonal filter banks have many attractive features, one eludes them: when restricted to real-coefficient, FIR filters, solutions that are both orthonormal and linear phase do not exist except for Haar filters. This is one of the key motivations for looking beyond the orthogonal case, as well as for the popularity of biorthogonal filter banks, especially in image processing. Similarly to the orthogonal case, we want to find out how to implement biorthogonal bases using filter banks, in particular, those having certain time and frequency localization properties. From Definition 1.42, we know that a system  $\{\varphi_k, \tilde{\varphi}_k\}$  constitutes a pair of biorthogonal bases of the Hilbert space  $\ell^2(\mathbb{Z})$ , if (1) they satisfy biorthogonality constraints (1.102):

$$\langle \varphi_k, \tilde{\varphi}_i \rangle = \delta_{k-i} \quad \leftrightarrow \quad \Phi \tilde{\Phi}^T = \tilde{\Phi} \Phi^T = I, \quad (7.59)$$

where  $\Phi$  is an infinite matrix having  $\varphi_k$  as its columns, while  $\tilde{\Phi}$  is an infinite matrix having  $\tilde{\varphi}_k$  as its columns; and (2) it is complete:

$$x = \sum_{k \in \mathbb{Z}} X_k \varphi_k = \Phi X = \sum_{k \in \mathbb{Z}} \tilde{X}_k \tilde{\varphi}_k = \tilde{\Phi} \tilde{X}, \quad (7.60)$$

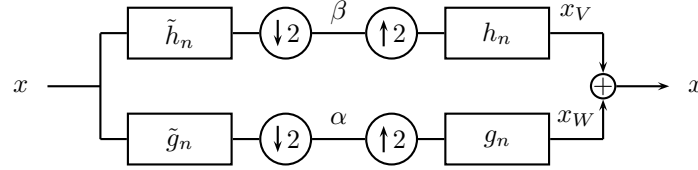
for all  $x \in \ell^2(\mathbb{Z})$ , where

$$X_k = \langle \tilde{\varphi}_k, x \rangle \quad \leftrightarrow \quad X = \tilde{\Phi}^T x, \quad \text{and} \quad \tilde{X}_k = \langle \varphi_k, x \rangle \quad \leftrightarrow \quad \tilde{X} = \Phi^T x.$$



## 7.4. Biorthogonal Two-Channel Filter Banks

593



**Figure 7.10:** A biorthogonal two-channel analysis/synthesis filter bank. The output is the sum of the lowpass approximation  $x_V$  and its highpass counterpart  $x_W$ .

It is not a stretch now to imagine that, similarly to the orthogonal case, we are looking for two template basis sequences—a lowpass/highpass pair  $g$  and  $h$ , and a dual pair  $\tilde{g}$  and  $\tilde{h}$  so that the biorthogonality constraints (7.59) are satisfied. Under the right circumstances described in this section, such a filter bank will compute a biorthogonal expansion. Assume that indeed, we are computing such an expansion. Start from the reconstructed output as in Figure 7.10:

$$x = x_V + x_W = \sum_{k \in \mathbb{Z}} \alpha_k g_{n-2k} + \sum_{k \in \mathbb{Z}} \beta_k h_{n-2k},$$

or

$$\underbrace{\begin{bmatrix} \vdots \\ x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \end{bmatrix}}_x = \underbrace{\begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & g_0 & h_0 & 0 & 0 & 0 & 0 & \dots \\ \dots & g_1 & h_1 & 0 & 0 & 0 & 0 & \dots \\ \dots & g_2 & h_2 & g_0 & h_0 & 0 & 0 & \dots \\ \dots & g_3 & h_3 & g_1 & h_1 & 0 & 0 & \dots \\ \dots & g_4 & h_4 & g_2 & h_2 & g_0 & h_0 & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{\Phi} \underbrace{\begin{bmatrix} \vdots \\ \alpha_0 \\ \beta_0 \\ \alpha_1 \\ \beta_1 \\ \alpha_2 \\ \vdots \end{bmatrix}}_X = \Phi X, \quad (7.61)$$

exactly the same as (7.7). As in (7.7),  $g_{n-2k}$  and  $h_{n-2k}$  are the impulse responses of the synthesis filters  $g$  and  $h$  shifted by  $2k$ , and  $\alpha_k$  and  $\beta_k$  are the outputs of the analysis filter bank downsampled by 2. The basis sequences are the columns of

$$\Phi = \{\varphi_k\}_{k \in \mathbb{Z}} = \{\varphi_{2k}, \varphi_{2k+1}\}_{k \in \mathbb{Z}} = \{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}, \quad (7.62)$$

that is, the even-indexed basis sequences are the impulse responses of the synthesis lowpass filter and its even shifts, while the odd-indexed basis sequences are the impulse responses of the synthesis highpass filter and its even shifts.

So far, the analysis has been identical to that of orthogonal filter banks; we repeated it here for emphasis. Since we are implementing a biorthogonal expansion, the transform coefficients  $\alpha_k$  and  $\beta_k$  are inner products between the dual basis

sequences and the input sequence:  $\alpha_k = \langle x, \tilde{\varphi}_{2k} \rangle$ ,  $\beta_k = \langle x, \tilde{\varphi}_{2k+1} \rangle$ . From (2.61a),

$$\begin{aligned} \alpha_k &= \langle x, \tilde{\varphi}_{2k} \rangle = \langle x_n, \tilde{g}_{2k-n} \rangle_n = \tilde{g}_{n-2k} * x & \leftrightarrow & \alpha = \tilde{\Phi}_g^T x, \\ \beta_k &= \langle x, \tilde{\varphi}_{2k+1} \rangle = \langle x_n, \tilde{h}_{2k-n} \rangle_n = \tilde{h}_{n-2k} * x & \leftrightarrow & \beta = \tilde{\Phi}_h^T x, \end{aligned}$$

that is, we can implement the computation of the expansion coefficients  $\alpha_k$  and  $\beta_k$  using convolutions, exactly as in the orthogonal case. We finally get

$$X = \tilde{\Phi}^T x.$$

From above, we see that the dual basis sequences are

$$\tilde{\Phi} = \{\tilde{\varphi}_k\}_{k \in \mathbb{Z}} = \{\tilde{\varphi}_{2k}, \tilde{\varphi}_{2k+1}\}_{k \in \mathbb{Z}} = \{\tilde{g}_{2k-n}, \tilde{h}_{2k-n}\}_{k \in \mathbb{Z}}, \quad (7.63)$$

that is, the even-indexed dual basis sequences are the shift-reversed impulse responses of the analysis lowpass filter and its even shifts, while the odd-indexed basis sequences are the shift-reversed impulse responses of the analysis highpass filter and its even shifts.

We stress again that the basis sequences of  $\Phi$  are synthesis filters' impulse responses and their even shifts, while the basis sequences of  $\tilde{\Phi}$  are the *shift-reversed* analysis filters' impulse responses and their even shifts. This shift reversal comes from the fact that we are implementing our inner product using a convolution. Note also that  $\Phi$  and  $\tilde{\Phi}$  are completely interchangeable.

As opposed to the three orthonormality relations (7.8), here we have four biorthogonality relations, visualized in Figure 7.11:

$$\langle g_n, \tilde{g}_{2k-n} \rangle_n = \delta_k, \quad (7.64a)$$

$$\langle h_n, \tilde{h}_{2k-n} \rangle_n = \delta_k, \quad (7.64b)$$

$$\langle h_n, \tilde{g}_{2k-n} \rangle_n = 0, \quad (7.64c)$$

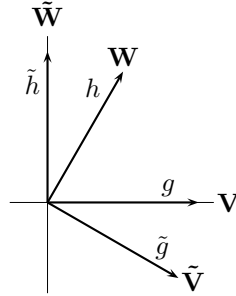
$$\langle g_n, \tilde{h}_{2k-n} \rangle_n = 0. \quad (7.64d)$$

The purpose of this section is to explore the family of impulse responses  $\{g, h\}$  and their duals  $\{\tilde{g}, \tilde{h}\}$  so as to satisfy the biorthogonality constraints. This family is much larger than the orthonormal family, and will contain symmetric/antisymmetric solutions, on which we will focus.

### 7.4.1 A Single Channel and Its Properties

As we have done for the orthogonal case, we first discuss channels in isolation and determine what they need to satisfy. Figure 7.12 shows the biorthogonal lowpass channel, projecting the input  $x$  onto its lowpass approximation  $x_V$ . That lowpass approximation  $x_V$  can be expressed identically to (7.12a):

$$x_V = \sum_{k \in \mathbb{Z}} \alpha_k g_{n-2k}. \quad (7.65a)$$



**Figure 7.11:** In a biorthogonal basis,  $\tilde{g}$  is orthogonal to  $h$ , and  $\tilde{h}$  is orthogonal to  $g$ . Then,  $\tilde{g}$  and  $\tilde{h}$  are normalized so that the inner products with their duals are 1.

The highpass channel follows the lowpass exactly, substituting  $h$  for  $g$ ,  $\tilde{h}$  for  $\tilde{g}$ , and  $x_W$  for  $x_V$  (see Figure 7.12). The highpass approximation  $x_W$  is

$$x_W = \sum_{k \in \mathbb{Z}} \beta_k h_{n-2k}. \quad (7.65b)$$

**Biorthogonality of the Lowpass Filters** Since we started with a pair of biorthogonal bases,  $\{g_{n-2k}, \tilde{g}_{2k-n}\}_{k \in \mathbb{Z}}$  satisfy biorthogonality relations (7.64a). Similarly to the orthogonal case, these can be expressed in various domains as:

$$\begin{array}{ll} \langle g_n, \tilde{g}_{2k-n} \rangle_n = \delta_k & \begin{array}{l} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} \end{array} \quad \begin{array}{l} D_2 \tilde{G} G U_2 = I \\ G(z) \tilde{G}(z) + G(-z) \tilde{G}(-z) = 2 \\ G(e^{j\omega}) \tilde{G}(e^{j\omega}) + G(e^{j(\omega+\pi)}) \tilde{G}(e^{j(\omega+\pi)}) = 2 \end{array} \quad (7.66)$$

In the matrix view, we have used linear operators (infinite matrices) as we did for the orthogonal case; it expresses the fact that the columns of  $GU_2$  are orthogonal to the rows of  $D_2 \tilde{G}$ . The  $z$ -transform expression is often the defining equation of a biorthogonal filter bank, where  $G(z)$  and  $\tilde{G}(z)$  are not causal in general.

#### Biorthogonality of the Highpass Filters

$$\begin{array}{ll} \langle h_n, \tilde{h}_{2k-n} \rangle_n = \delta_k & \begin{array}{l} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} \end{array} \quad \begin{array}{l} D_2 \tilde{H} H U_2 = I \\ H(z) \tilde{H}(z) + H(-z) \tilde{H}(-z) = 2 \\ H(e^{j\omega}) \tilde{H}(e^{j\omega}) + H(e^{j(\omega+\pi)}) \tilde{H}(e^{j(\omega+\pi)}) = 2 \end{array} \quad (7.67)$$

**Deterministic Crosscorrelation of the Lowpass Filters** In the orthogonal case, we rephrased relations as in (7.66) in terms of the deterministic autocorrelation of  $g$ ;

**Lowpass Channel in a Two-Channel Biorthogonal Filter Bank****Lowpass filters**

Original domain	$g_n, \tilde{g}_n$	$\langle g_n, \tilde{g}_{2k-n} \rangle_n = \delta_k$
Matrix domain	$G, \tilde{G}$	$D_2 \tilde{G} G U_2 = I$
$z$ domain	$G(z), \tilde{G}(z)$	$G(z) \tilde{G}(z) + G(-z) \tilde{G}(-z) = 2$
DTFT domain	$G(e^{j\omega}), \tilde{G}(e^{j\omega})$	$G(e^{j\omega}) \tilde{G}(e^{j\omega}) + G(e^{j(\omega+\pi)}) \tilde{G}(e^{j(\omega+\pi)}) = 2$
Polyphase domain	$G(z) = G_0(z^2) + z^{-1} G_1(z^2)$ $\tilde{G}(z) = \tilde{G}_0(z^2) + z \tilde{G}_1(z^2)$	$G_0(z) \tilde{G}_0(z) + G_1(z) \tilde{G}_1(z) = 1$

**Deterministic crosscorrelation**

Original domain	$c_n = \langle g_k, \tilde{g}_{k+n} \rangle_k$	
Matrix domain	$C = \tilde{G} G$	$D_2 C U_2 = I$
$z$ domain	$C(z) = G(z) \tilde{G}(z^{-1})$	$C(z) + C(-z) = 2$
DTFT domain	$C(e^{j\omega}) = G(e^{j\omega}) \tilde{G}(e^{j\omega})$	$C(e^{j\omega}) + C(e^{j(\omega+\pi)}) = 2$

**Oblique projection onto smooth space**  $V = \overline{\text{span}}(\{g_{n-2k}\}_{k \in \mathbb{Z}})$   
 $x_V = P_V x$   $P_V = G U_2 D_2 \tilde{G}$

**Table 7.4:** Properties of the lowpass channel in a biorthogonal two-channel filter bank. Properties for the highpass channel are analogous. With  $\tilde{g}_n = g_{-n}$ , or,  $\tilde{G}(z) = G(z^{-1})$  in the  $z$ -transform domain, the relations in this table reduce to those in Table 7.1 for the orthogonal two-channel filter bank.

here, as we have two sequences  $g$  and  $\tilde{g}$ , we express it in terms of the deterministic crosscorrelation of  $g$  and  $\tilde{g}$ , (2.99):

$$\begin{array}{ccc}
 & \text{Matrix View} & \\
 \langle g_n, \tilde{g}_{2k-n} \rangle_n = c_{2k} = \delta_k & \begin{array}{c} \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_2 C U_2 = I \\ C(z) + C(-z) = 2 \\ C(e^{j\omega}) + C(e^{j(\omega+\pi)}) = 2 \end{array} \quad (7.68)
 \end{array}$$

In the above,  $C = \tilde{G} G$  is a Toeplitz matrix with element  $c_{\pm k}$  on the  $k$ th diagonal left/right from the main diagonal (see (1.228)). While this deterministic crosscorrelation will be used for design as in the orthogonal case, unlike in the orthogonal case: (1)  $C(z)$  does not have to be symmetric; (2)  $C(e^{j\omega})$  does not have to be positive; and (3) any factorization of  $C(z)$  leads to a valid solution, that is, the roots of  $C(z)$  can be arbitrarily assigned to  $\tilde{G}(z)$  and  $G(z)$ .

**Deterministic Crosscorrelation of the Highpass Filters**

$$\begin{array}{ccc}
 & \text{Matrix View} & \\
 \langle h_n, \tilde{h}_{2k-n} \rangle_n = c_{2k} = \delta_k & \begin{array}{c} \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_2 C U_2 = I \\ C(z) + C(-z) = 2 \\ C(e^{j\omega}) + C(e^{j(\omega+\pi)}) = 2 \end{array} \quad (7.69)
 \end{array}$$

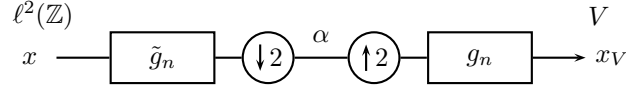


Figure 7.12: The biorthogonal lowpass channel.

**Projection Property of the Lowpass Channel** We now look at the lowpass channel as a composition of linear operators:

$$x_V = P_V x = GU_2 D_2 \tilde{G} x. \quad (7.70)$$

While  $P_V$  is a projection, it is not an orthogonal projection:

$$\begin{aligned} P_V^2 &= (GU_2 \underbrace{D_2 \tilde{G}}_I) (GU_2 D_2 \tilde{G}) = GU_2 D_2 \tilde{G} = P_V, \\ P_V^T &= (GU_2 D_2 \tilde{G})^T = \tilde{G}^T (U_2 D_2)^T G^T = \tilde{G}^T U_2 D_2 G^T \neq P_V. \end{aligned}$$

Indeed,  $P_V$  is a projection operator (it is idempotent), but it is not orthogonal (it is not self-adjoint). Its range is as in the orthogonal case:

$$V = \overline{\text{span}}(\{g_{n-2k}\}_{k \in \mathbb{Z}}). \quad (7.71)$$

Note the interchangeable roles of  $\tilde{g}$  and  $g$ . When  $g$  is used in the synthesis, then  $x_V$  lives in the above span, while if  $\tilde{g}$  is used, it lives in the span of  $\{\tilde{g}_{n-2k}\}_{k \in \mathbb{Z}}$ . The summary of properties of the lowpass channel is given in Table 7.4.

**Projection Property of the Highpass Channel** The highpass projection operator  $P_W$  is:

$$x_W = P_W x = HU_2 D_2 \tilde{H} x; \quad (7.72)$$

again a projection operator (it is idempotent), but not orthogonal (it is not self-adjoint) the same way as for  $P_V$ . Its range is:

$$W = \overline{\text{span}}(\{h_{n-2k}\}_{k \in \mathbb{Z}}). \quad (7.73)$$

### 7.4.2 Complementary Channels and Their Properties

Following the path set during the analysis of orthogonal filter banks, we now discuss what the two channels have to satisfy with respect to each other to build a biorthogonal filter bank. Given a pair of filters  $g$  and  $\tilde{g}$  satisfying (7.66), how can we choose  $h$  and  $\tilde{h}$  to complete the biorthogonal filter bank and thus implement a biorthogonal basis expansion? The sets of basis and dual basis sequences  $\{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}$  and  $\{\tilde{g}_{2k-n}, \tilde{h}_{2k-n}\}_{k \in \mathbb{Z}}$  must satisfy (7.64). We have already used (7.64a) in (7.66) and similarly for the highpass sequences in (7.67). What is left to use is that these lowpass and highpass sequences are orthogonal to each other as in (7.64c)–(7.64d):

**Orthogonality of the Lowpass and Highpass Filters**

$$\begin{array}{ccc}
\langle h_n, \tilde{g}_{2k-n} \rangle_n = 0 & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_2 \tilde{G} H U_2 = 0 \\ H(z) \tilde{G}(z) + H(-z) \tilde{G}(-z) = 0 \\ H(e^{j\omega}) \tilde{G}(e^{j\omega}) + H(e^{j(\omega+\pi)}) \tilde{G}(e^{j(\omega+\pi)}) = 0 \end{array} \\
& & (7.74a)
\end{array}$$

and similarly for  $g$  and  $\tilde{h}$ :

$$\begin{array}{ccc}
\langle g_n, \tilde{h}_{2k-n} \rangle_n = 0 & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_2 \tilde{H} G U_2 = 0 \\ G(z) \tilde{H}(z) + G(-z) \tilde{H}(-z) = 0 \\ G(e^{j\omega}) \tilde{H}(e^{j\omega}) + G(e^{j(\omega+\pi)}) \tilde{H}(e^{j(\omega+\pi)}) = 0 \end{array} \\
& & (7.74b)
\end{array}$$

**7.4.3 Biorthogonal Two-Channel Filter Bank**

We now pull together what we have developed for biorthogonal filter banks. The following result gives one possible example of a biorthogonal filter bank, inspired by the orthogonal case. We choose the highpass synthesis filter as a modulated version of the lowpass, together with an odd shift. However, because of biorthogonality, it is the analysis lowpass that comes into play.

**THEOREM 7.7 (BIORTHOGONAL TWO-CHANNEL FILTER BANK)** Given are two FIR filters  $g$  and  $\tilde{g}$  of even length  $L = 2\ell$ ,  $\ell \in \mathbb{Z}^+$ , orthogonal to each other and their even shifts as in (7.66). Choose

$$h_n = (-1)^n \tilde{g}_{n-2\ell+1} \quad \xleftrightarrow{\text{ZT}} \quad H(z) = -z^{-L+1} \tilde{G}(-z) \quad (7.75a)$$

$$\tilde{h}_n = (-1)^n g_{n+2\ell-1} \quad \xleftrightarrow{\text{ZT}} \quad \tilde{H}(z) = -z^{L-1} G(-z) \quad (7.75b)$$

Then, sets  $\{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}$  and  $\{\tilde{g}_{2k-n}, \tilde{h}_{2k-n}\}_{k \in \mathbb{Z}}$  are a pair of biorthogonal bases for  $\ell^2(\mathbb{Z})$ , implemented by a biorthogonal filter bank specified by analysis filters  $\{\tilde{g}, \tilde{h}\}$  and synthesis filters  $\{g, h\}$ .

*Proof.* To prove the theorem, we must prove that (i)  $\{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}$  and  $\{\tilde{g}_{2k-n}, \tilde{h}_{2k-n}\}_{k \in \mathbb{Z}}$  are biorthogonal sets and (ii) they are complete.

- (i) To prove that  $\{g_{n-2k}, h_{n-2k}\}_{k \in \mathbb{Z}}$  and  $\{\tilde{g}_{2k-n}, \tilde{h}_{2k-n}\}_{k \in \mathbb{Z}}$  are biorthogonal sets, we must prove (7.64). The first condition, (7.64a), is satisfied by assumption. To prove the second, (7.64b), that is,  $h$  is orthogonal to  $\tilde{h}$  and its even shifts, we must prove one of the conditions in (7.67). The definitions of  $h$  and  $\tilde{h}$  in (7.75) imply

$$H(z) \tilde{H}(z) = G(-z) \tilde{G}(-z) \quad (7.76)$$

and thus,

$$H(z) \tilde{H}(z) + H(-z) \tilde{H}(-z) = G(-z) \tilde{G}(-z) + G(z) \tilde{G}(z) \stackrel{(a)}{=} 2,$$

## 7.4. Biorthogonal Two-Channel Filter Banks

599

where (a) follows from (7.66).

To prove (7.64c)–(7.64d), we must prove one of the conditions in (7.74a)–(7.74b), respectively. We prove (7.64c), (7.64d) follows similarly.

$$\begin{aligned} H(z)\tilde{G}(z) + H(-z)\tilde{G}(-z) &\stackrel{(a)}{=} -z^{L-1}\tilde{G}(-z)\tilde{G}(z) - (-1)^{-L+1}z^{L-1}\tilde{G}(z)\tilde{G}(-z) \\ &\stackrel{(b)}{=} -z^{L-1}G(-z)\tilde{G}(z) + z^{L-1}\tilde{G}(z)\tilde{G}(-z) = 0, \end{aligned}$$

where (a) follows from (7.75a); and (b)  $L = 2\ell$  even.

- (ii) To prove completeness, we prove that perfect reconstruction holds for any  $x \in \ell^2(\mathbb{Z})$ . What we do is find  $z$ -domain expressions for  $X_V(z)$  and  $X_W(z)$  and prove they sum up to  $X(z)$ . We start with the lowpass branch. The proof proceeds as in the orthogonal case.

$$X_V(z) = \frac{1}{2}G(z) \left[ \tilde{G}(z)X(z) + \tilde{G}(-z)X(-z) \right], \quad (7.77a)$$

$$X_W(z) = \frac{1}{2}H(z) \left[ \tilde{H}(z)X(z) + \tilde{H}(-z)X(-z) \right]. \quad (7.77b)$$

The output of the filter bank is the sum of  $x_V$  and  $x_W$ :

$$\begin{aligned} X_V(z) + X_W(z) &= \frac{1}{2} \underbrace{\left[ G(z)\tilde{G}(-z) + H(z)\tilde{H}(-z) \right]}_{S(z)} X(-z) \\ &\quad + \frac{1}{2} \underbrace{\left[ G(z)\tilde{G}(z) + H(z)\tilde{H}(z) \right]}_{T(z)} X(z). \end{aligned} \quad (7.78)$$

Substituting (7.75) into the above equation, we get:

$$\begin{aligned} S(z) &= G(z)\tilde{G}(-z) + H(z)\tilde{H}(-z) \\ &\stackrel{(a)}{=} G(z)\tilde{G}(z) + \left[ -z^{-L+1}\tilde{G}(-z) \right] \left[ -(-z)^{L-1}G(z) \right] \\ &= \left[ 1 + (-1)^{L+1} \right] G(z)\tilde{G}(-z) \stackrel{(b)}{=} 0, \\ T(z) &= G(z)\tilde{G}(z) + H(z)\tilde{H}(z) \\ &\stackrel{(c)}{=} G(z)\tilde{G}(z) + \tilde{G}(-z)G(-z) \stackrel{(d)}{=} 2, \end{aligned}$$

where (a) follows from (7.75); (b) from  $L = 2\ell$  even; (c) from (7.75); and (d) from (7.66). Substituting this back into (7.78), we get

$$X_V(z) + X_W(z) = X(z), \quad (7.79)$$

proving perfect reconstruction, or, in other words, the assertion in the theorem statement that the expansion can be implemented by a biorthogonal filter bank.

Note that we could have also expressed our design problem based on the synthesis (analysis) filters only.

Unlike the orthogonal case, the approximation spaces  $V$  and  $W$  are not orthogonal anymore, and therefore, there exist dual spaces  $\tilde{V}$  and  $\tilde{W}$  spanned by  $\tilde{g}_{-n}$  and  $\tilde{h}_{-n}$  and their even shifts. However,  $V$  is orthogonal to  $\tilde{W}$  and  $W$  is orthogonal to  $\tilde{V}$ . This was schematically shown in Figure 7.11. Table 7.10 summarizes various properties of biorthogonal, two-channel filter banks we covered until now.

### 7.4.4 Polyphase View of Biorthogonal Filter Banks

We have already seen how polyphase analysis of orthogonal filter banks adds to the analysis toolbox. We now give a brief account of important polyphase notions when dealing with biorthogonal filter banks. First, recall from (7.33) that the polyphase matrix of the synthesis bank is given by<sup>110</sup>

$$\Phi_p(z) = \begin{bmatrix} G_0(z) & H_0(z) \\ G_1(z) & H_1(z) \end{bmatrix}, \quad \begin{aligned} G(z) &= G_0(z) + z^{-1}G_1(z), \\ H(z) &= H_0(z) + z^{-1}H_1(z). \end{aligned} \quad (7.80a)$$

By the same token, the polyphase matrix of the analysis bank is given by

$$\tilde{\Phi}_p(z) = \begin{bmatrix} \tilde{G}_0(z) & \tilde{H}_0(z) \\ \tilde{G}_1(z) & \tilde{H}_1(z) \end{bmatrix}, \quad \begin{aligned} \tilde{G}(z) &= \tilde{G}_0(z) + z\tilde{G}_1(z), \\ \tilde{H}(z) &= \tilde{H}_0(z) + z\tilde{H}_1(z). \end{aligned} \quad (7.80b)$$

Remember that the different polyphase decompositions of the analysis and synthesis filters are a matter of a carefully chosen convention.

For a biorthogonal filter bank to implement a biorthogonal expansion, the following must be satisfied:

$$\Phi_p(z) \tilde{\Phi}_p^T(z) = I. \quad (7.81)$$

From this,

$$\tilde{\Phi}_p(z) = (\Phi_p^T(z))^{-1} = \frac{1}{\det \Phi_p(z)} \begin{bmatrix} H_1(z) & -H_0(z) \\ -G_1(z) & G_0(z) \end{bmatrix}. \quad (7.82)$$

Since all the matrix entries are FIR, for the analysis to be FIR as well,  $\det \Phi_p(z)$  must be a monomial, that is:

$$\det \Phi_p(z) = G_0(z)H_1(z) - G_1(z)H_0(z) = z^{-k}. \quad (7.83)$$

In the above, we have implicitly assumed that  $\Phi_p(z)$  was invertible, that is, its columns are linearly independent. This can be rephrased in filter bank terms by stating when, given  $G(z)$ , it is possible to find  $H(z)$  such that it leads to a perfect reconstruction biorthogonal filter bank. Such a filter  $H(z)$  will be called a *complementary filter*.

**PROPOSITION 7.8 (COMPLEMENTARY FILTERS)** Given a causal FIR filter  $G(z)$ , there exists a complementary FIR filter  $H(z)$ , if and only if the polyphase components of  $G(z)$  are coprime (except for possible zeros at  $z = \infty$ ).

*Proof.* We just saw that a necessary and sufficient condition for perfect FIR reconstruction is that  $\det(\Phi_p(z))$  be a monomial. Thus, coprimeness is obviously necessary, since if there were a common factor between  $G_0(z)$  and  $G_1(z)$ , it would show up in the determinant.

<sup>110</sup>When we say *polyphase matrix*, we will mean the polyphase matrix of the synthesis bank; for the analysis bank, we will explicitly state *analysis polyphase matrix*.



## 7.4. Biorthogonal Two-Channel Filter Banks

601

Sufficiency follows from the Bézout's identity (2.282) that says that given two coprime polynomials  $a(z)$  and  $b(z)$ , the equation  $a(z)p(z) + b(z)q(z) = c(z)$  has a solution  $p(z), q(z)$ . Fixing  $a(z) = G_0(z)$ ,  $b(z) = G_1(z)$  and  $c(z) = z^{-k}$ , we see that Bézout's identity is equal to (7.83), and thus guarantees a solution  $p(z) = H_0(z)$  and  $q(z) = H_1(z)$ , that is, a complementary filter  $H(z)$ .

Note that the coprimeness of  $G_0(z)$  and  $G_1(z)$  is equivalent to  $G(z)$  not having any zero pairs  $\{z_0, -z_0\}$ . This can be used to prove that the binomial filter  $G(z) = (1 + z^{-1})^N$  always has a complementary filter (see Exercise 7.12).

The counterpart to Theorem 7.3 and Corollary 7.4 for orthogonal filter banks are the following theorem and corollary for the biorthogonal ones (we state these without proof):

**THEOREM 7.9 (POSITIVE DEFINITE MATRIX AND BIORTHOGONAL BASIS)** Given a filter bank implementing a biorthogonal basis for  $\ell^2(\mathbb{Z})$  and its associated polyphase matrix  $\Phi_p(e^{j\omega})$ , then  $\Phi_p(e^{j\omega})\Phi_p^T(e^{-j\omega})$  is positive definite.

**COROLLARY 7.10 (FILTERED DETERMINISTIC AUTOCORRELATION MATRIX IS POSITIVE SEMIDEFINITE)** Given is a  $2 \times 2$  polyphase matrix  $\Phi_p(e^{j\omega})$  such that  $\Phi_p(e^{j\omega})\Phi_p^T(e^{-j\omega})$ . Then the filtered deterministic autocorrelation matrix,  $A_{p,\alpha}(e^{j\omega})$ , is positive semidefinite.

## 7.4.5 Linear-Phase Two-Channel Filter Banks

We started this section by saying that one of the reasons we go through the trouble of analyzing and constructing two-channel biorthogonal filter banks is because they allow us to obtain real-coefficient FIR filters with linear phase.<sup>111</sup> Thus, we now do just that: we build perfect reconstruction filter banks where all the filters involved are linear phase. Linear-phase filters were defined in (2.106).

As was true for orthogonal filters, not all lengths of filters are possible if we want to have a linear-phase filter bank. This is summarized in the following proposition, the proof of which is left as Exercise 7.14:

**PROPOSITION 7.11** In a two-channel, perfect reconstruction filter bank where all filters are linear phase, the synthesis filters have one of the following forms:

- (i) Both filters are odd-length symmetric, the lengths differing by an odd multiple of 2.
- (ii) One filter is symmetric and the other is antisymmetric; both lengths are even, and are equal or differ by an even multiple of 2.

<sup>111</sup>If we allow filters to have complex-valued coefficients or if we lift the restriction of two channels, linear phase and orthogonality can be satisfied simultaneously.

- (iii) One filter is of odd length, the other one of even length; both have all zeros on the unit circle. Either both filters are symmetric, or one is symmetric and the other one is antisymmetric.

Our next task is to show that indeed, it is not possible to have an orthogonal filter bank with linear-phase filters if we restrict ourselves to the two-channel, FIR, real-coefficient case:

**PROPOSITION 7.12** The only two-channel perfect reconstruction orthogonal filter bank with real-coefficient FIR linear-phase filters is the Haar filter bank.

*Proof.* In orthogonal filter banks, (7.40)–(7.41) hold, and the filters are of even length. Therefore, following Proposition 7.11, one filter is symmetric and the other antisymmetric. Take the symmetric one,  $G(z)$  for example,

$$\begin{aligned} G(z) &\stackrel{(a)}{=} G_0(z^2) + z^{-1}G_1(z^2) \\ &\stackrel{(b)}{=} z^{-L+1}G(z^{-1}) \stackrel{(c)}{=} z^{-L+1}(G_0(z^{-2}) + zG_1(z^{-2})) \\ &= z^{-L+2}G_1(z^{-2}) + z^{-1}(z^{-L+2}G_0(z^{-2})), \end{aligned}$$

where (a) and (c) follow from (7.32), and (b) from (2.152). This further means that for the polyphase components, the following hold:

$$G_0(z) = z^{-L/2+1}G_1(z^{-1}), \quad G_1(z) = z^{-L/2+1}G_0(z^{-1}). \quad (7.84)$$

Substituting (7.84) into (7.40) we obtain

$$G_0(z) G_0(z^{-1}) = \frac{1}{2}.$$

The only FIR, real-coefficient polynomial satisfying the above is

$$G_0(z) = \frac{1}{\sqrt{2}}z^{-m}.$$

Performing a similar analysis for  $G_1(z)$ , we get that  $G_1(z) = (1/\sqrt{2})z^{-k}$ , and

$$G(z) = \frac{1}{\sqrt{2}}(z^{-2\ell} + z^{-2k-1}), \quad H(z) = G(-z),$$

yielding Haar filters ( $m = k = 0$ ) or trivial variations thereof.

While the outstanding features of the Haar filters make it a very special solution, Proposition 7.12 is a fundamentally negative result as the Haar filters have poor frequency localization and no polynomial reproduction capability.

## 7.5 Design of Biorthogonal Two-Channel Filter Banks

Given that biorthogonal filters are less constrained than their orthogonal counterparts, the design space is much more open. In both cases, one factors a Laurent polynomial<sup>112</sup>  $C(z)$  satisfying  $C(z) + C(-z) = 2$  as in (7.68). In the orthogonal

<sup>112</sup>A Laurent polynomial is a polynomial with both positive and negative powers, see Appendix 2.B.1.

case,  $C(z)$  was a deterministic autocorrelation, while in the biorthogonal case, it is a deterministic crosscorrelation and thus more general. In addition, the orthogonal case requires spectral factorization (square root), while in the biorthogonal case, any factorization will do. While the factorization method is not the only approach, it is the most common. Other approaches include the complementary filter design method and the lifting design method. In the former, a desired filter is complemented so as to obtain a perfect reconstruction filter bank. In the latter, a structure akin to a lattice is used to guarantee perfect reconstruction as well as other desirable properties.

### 7.5.1 Factorization Design

From (7.66)–(7.68),  $C(z)$  satisfying  $C(z) + C(-z) = 2$  can be factored into

$$C(z) = G(z)\tilde{G}(z),$$

where  $G(z)$  is the synthesis and  $\tilde{G}(z)$  the analysis lowpass filter (or vice-versa, since the roles are dual). The most common designs use the same  $C(z)$  as those used in orthogonal filter banks, for example, those with a maximum number of zeros at  $z = -1$ , performing the factorization so that the resulting filters have linear phase.

**EXAMPLE 7.4 (BIORTHOGONAL FILTER BANK WITH LINEAR-PHASE FILTERS)** We reconsider Example 7.3, in particular  $C(z)$  given by

$$C(z) = (1 + z^{-1})^2 (1 + z)^2 \frac{1}{4} \left( -\frac{1}{4}z^{-1} + 1 - \frac{1}{4}z \right),$$

which satisfies  $C(z) + C(-z) = 2$  by construction. This also means it satisfies (7.66) for any factorization of  $C(z)$  into  $\tilde{G}(z)G(z)$ . Note that we can add factors  $z$  or  $z^{-1}$  in one filter, as long as we cancel it in the other; this is useful for obtaining purely causal/anticausal solutions.

One possible factorization is

$$\begin{aligned} G(z) &= z^{-1} (1 + z^{-1})^2 (1 + z) = (1 + z^{-1})^3 = 1 + 3z^{-1} + 3z^{-2} + z^{-3}, \\ \tilde{G}(z) &= z(1 + z) \frac{1}{4} \left( -\frac{1}{4}z^{-1} + 1 - \frac{1}{4}z \right) = \frac{1}{16} (-1 + 3z + 3z^2 - z^3). \end{aligned}$$

The other filters follow from (7.75), with  $L = 2\ell = 4$ :

$$\begin{aligned} H(z) &= -z^{-3} \frac{1}{16} (-1 - 3z + 3z^2 + z^3) = \frac{1}{16} (-1 - 3z^{-1} + 3z^{-2} + z^{-3}), \\ \tilde{H}(z) &= -z^3 (1 - 3z^{-1} + 3z^{-2} - z^{-3}) = 1 - 3z + 3z^2 - z^3. \end{aligned}$$

The lowpass filters are both symmetric, while the highpass ones are antisymmetric. As  $\tilde{H}(z)$  has three zero moments,  $G(z)$  can reproduce polynomials up to degree 2, since such sequences go through the lowpass channel only.

Another possible factorization is

$$\begin{aligned} G(z) &= (1 + z^{-1})^2 (1 + z)^2 = z^{-2} + 4z^{-1} + 6 + 4z + z^2, \\ \tilde{G}(z) &= \frac{1}{4} \left( -\frac{1}{4}z^{-1} + 1 - \frac{1}{4}z \right) = \frac{1}{16} (-z^{-1} + 4 - z), \end{aligned}$$

where both lowpass filters are symmetric and zero phase. The highpass filters are (with  $L = 0$ ):

$$\begin{aligned} H(z) &= -\frac{1}{16}z(z^{-1} + 4 + z), \\ \tilde{H}(z) &= -z^{-1}(z^{-2} - 4z^{-1} + 6 - 4z + z^2), \end{aligned}$$

which are also symmetric, but with a phase delay of  $\pm 1$  sample.

The zeros at  $z = -1$  in the synthesis lowpass filter become, following (7.75b), zeros at  $z = 1$  in the analysis highpass filter. Therefore, many popular biorthogonal filters come from symmetric factorizations of  $C(z)$  with a maximum number of zeros at  $z = -1$ .

**EXAMPLE 7.5 (DESIGN OF THE 9/7 FILTER PAIR)** The next higher-order  $C(z)$  with a maximum number of zeros at  $z = -1$  is of the form

$$C(z) = 2^{-8} (1 + z)^3 (1 + z^{-1})^3 (3z^2 - 18z + 38 - 18z^{-1} + 3z^{-2}).$$

One possible factorization yields the so-called *Daubechies 9/7* filter pair (see Table 7.5). These filters have odd length and even symmetry, and are part of the JPEG 2000 image compression standard.

Daubechies 9/7			LeGall 5/3	
$n$	$\tilde{g}_n$	$g_n$	$\tilde{g}_n$	$g_n$
0	0.60294901823635790	1.11508705245699400	3/4	1
$\pm 1$	0.26686411844287230	0.59127176311424700	1/4	1/2
$\pm 2$	-0.07822326652898785	-0.05754352622849957	-1/8	
$\pm 3$	-0.01686411844287495	-0.09127176311424948		
$\pm 4$	0.02674875741080976			

**Table 7.5:** Biorthogonal filters used in the still-image compression standard JPEG 2000. The lowpass filters are given; the highpass filters can be derived using (7.75a)–(7.75b). The first pair is from [5] and the second from [95].

## 7.5.2 Complementary Filter Design

Assume we have a desired synthesis lowpass filter  $G(z)$ . How can we find  $\tilde{G}(z)$  such that we obtain a perfect reconstruction biorthogonal filter bank? It suffices to

find  $\tilde{G}(z)$  so that (7.66) is satisfied, which, according to Proposition 7.8, can always be done if  $G(z)$  has coprime polyphase components. Then  $\tilde{G}(z)$  can be found by solving a linear system of equations.

EXAMPLE 7.6 (COMPLEMENTARY FILTER DESIGN) Suppose

$$G(z) = \frac{1}{2}z + 1 + \frac{1}{2}z^{-1} = \frac{1}{2}(1+z)(1+z^{-1}).$$

We would like to find  $\tilde{G}(z)$  such that  $C(z) = G(z)\tilde{G}(z)$  satisfies  $C(z) + C(-z) = 2$ . (It is easy to verify that the polyphase components of  $G(z)$  are coprime, so such a  $\tilde{G}(z)$  should exist. We exclude the trivial solution  $\tilde{G}(z) = 1$ ; it is of no interest as it has no frequency selectivity.) For a length-5 symmetric filter  $\tilde{G}(z) = cz^2 + bz + a + bz^{-1} + cz^{-2}$ , we get the following system of equations:

$$a + b = 1 \quad \text{and} \quad \frac{1}{2}b + c = 0.$$

To get a unique solution, we could, for example, impose that the filter have a zero at  $z = -1$ ,

$$a - 2b + 2c = 0,$$

leading to  $a = 6/8$ ,  $b = 2/8$ , and  $c = -1/8$ :

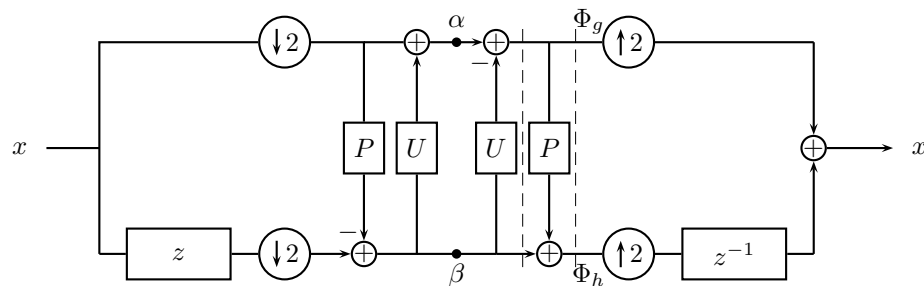
$$\tilde{G}(z) = \frac{1}{8}(-z^2 + 2z + 6 + 2z^{-1} - z^{-2}).$$

All coefficients of  $(g, \tilde{g})$  are integer multiples of  $1/8$ , making the analysis and synthesis exactly invertible even with finite-precision (binary) arithmetic. These filters are used in the JPEG 2000 image compression standard; see Table 7.5.

As can be seen from this example, the solution for the complementary filter is highly nonunique. Not only are there solutions of different lengths (in the case above, any length  $3 + 4m$ ,  $m \in \mathbb{N}$ , is possible), but even a given length has multiple solutions. It can be shown that this variety is given by the solutions of a Diophantine equation related to the polyphase components of the filter  $G(z)$ .

### 7.5.3 Lifting Design

We conclude this section with the design procedure based on lifting. While the original idea behind lifting was to build shift-varying perfect reconstruction filter banks, it has also become popular as it allows for building discrete-time bases with non-linear operations. The trivial filter bank to start lifting is the polyphase transform which splits the sequence into even- and odd-indexed components as in Figure 7.13. In the first lifting step, we use a prediction filter  $P$  to predict the odd coefficients from the even ones. The even coefficients remain unchanged, while the result of the prediction filter applied to the even coefficients is subtracted from the odd coefficients yielding the highpass coefficients. In the second step, we use an update filter  $U$  to update the even coefficients based on the previously computed highpass coefficients. We start with a simple example.



**Figure 7.13:** The lifting filter bank, with  $P$  and  $U$  predict and update operators, respectively.

EXAMPLE 7.7 (HAAR FILTER BANK OBTAINED BY LIFTING) The two polyphase components of  $x$  are  $x_0$  (even subsequence) and  $x_1$  (odd subsequence) as in (2.210). The purpose of the prediction operator  $P$  is to predict odd coefficients based on the even ones. The simplest prediction says that the odd coefficients are exactly the same as the even ones, that is  $p_n = \delta_n$ . The output of the highpass branch is thus the difference  $(\delta_n - \delta_{n-1})$ , a reasonable outcome. The purpose of the update operator  $U$  is to then update the even coefficients based on the newly computed odd ones. As we are looking for a lowpass-like version in the other branch, the easiest is to subtract half of this difference from the even sequence, leading to  $x_{0,n} - (x_{1,n} - x_{0,n})/2$ , that is, the average  $(x_{0,n} + x_{1,n})/2$ , again a reasonable output, but this time lowpass in nature. Within scaling, it is thus clear that the choice  $p_n = \delta_n$ ,  $u_n = (1/2)\delta_n$  leads to the Haar filter bank.

Let us now identify the polyphase matrix  $\Phi_p(z)$ :

$$\begin{aligned}\Phi_g(z) &= \alpha(z) - U(z)\beta(z), \\ \Phi_h(z) &= \beta(z) + P(z)\Phi_g(z) \\ &= \beta(z) + P(z)(\alpha(z) - U(z)\beta(z)) \\ &= P(z)\alpha(z) + (1 - P(z)U(z))\beta(z),\end{aligned}$$

which we can write as

$$\begin{bmatrix} \Phi_g(z) \\ \Phi_h(z) \end{bmatrix} = \begin{bmatrix} 1 & -U(z) \\ P(z) & 1 - P(z)U(z) \end{bmatrix} \begin{bmatrix} \alpha(z) \\ \beta(z) \end{bmatrix} = \Phi_p(z) \begin{bmatrix} \alpha(z) \\ \beta(z) \end{bmatrix}. \quad (7.85)$$

On the analysis side,  $\tilde{\Phi}_p(z)$  is:

$$\tilde{\Phi}_p(z) = (\Phi_p^T(z))^{-1} = \begin{bmatrix} 1 - P(z)U(z) & -P(z) \\ U(z) & 1 \end{bmatrix}. \quad (7.86)$$

As the  $\det(\Phi_p(z)) = 1$ , the inverse of  $\Phi_p(z)$  does not involve actual inversion, one of the reasons why this technique is popular. Moreover, we can write  $\Phi_p$  as

$$\Phi_p(z) = \begin{bmatrix} 1 & -U(z) \\ P(z) & 1 - P(z)U(z) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ P(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & -U(z) \\ 0 & 1 \end{bmatrix}, \quad (7.87)$$

decomposing  $\Phi_p(z)$  into a sequence of lower/upper triangular matrices—*lifting steps*. What we also see is that the inverse of each matrix of the form:

$$\begin{bmatrix} 1 & 0 \\ M & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 \\ -M & 1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1 & M \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & -M \\ 0 & 1 \end{bmatrix},$$

meaning to invert these one needs only reverse the sequence of operations as shown in Figure 7.13. This is why this scheme allows for nonlinear operations; if  $M$  is nonlinear, its inversion amounts to simply reversing the sign in the matrix.

## 7.6 Two-Channel Filter Banks with Stochastic Inputs

Our discussion so far assumed we are dealing with deterministic sequences as inputs into our filter bank, most often those with finite energy. If the input into our filter bank is stochastic, then we must use the tools developed in Chapter 2, Section 2.8. The periodic shift variance for deterministic systems has its counterpart in wide-sense cyclostationarity. The notions of energy spectral density (2.96) (DTFT of the deterministic autocorrelation) and energy (2.98) have their counterparts in the notions of power spectral density (2.232) (DTFT of the stochastic autocorrelation) and power (2.233). We now briefly discuss the effects of a filter bank on an input WSS sequence.

Until now, we have seen various ways of characterizing systems with deterministic and stochastic inputs, among others via the deterministic and stochastic autocorrelations.

In a single-input single-output system:

- (i) For a deterministic sequence, its autocorrelation is Hermitian symmetric (see (2.16)) and can be factored as in (2.96), (2.142), that is, it is nonnegative on the unit circle. It is sometimes called energy spectral density.
- (ii) For a WSS sequence, the counterpart to the deterministic autocorrelation is the power spectral density given in (2.232).

In a multiple-input multiple-output system, such as a filter banks, where the multiple inputs are naturally polyphase components of the input sequence:

- (i) For a deterministic sequence, we have a matrix autocorrelation of the vector of polyphase components  $[x_0 \ x_1]^T$ , given by (2.214). In particular, we have seen it for a the vector of expansion coefficient sequences  $[\alpha \ \beta]^T$  in two-channel filter bank earlier in this chapter, in (7.43).
- (ii) For a WSS sequence  $x$ , we can also look at the matrix of power spectral densities of the polyphase components  $[x_0 \ x_1]^T$  as in (2.245). In what follows, we analyze that matrix for the vector of expansion coefficient sequences  $[\alpha \ \beta]^T$ .

**Filter-Bank Optimization Based on Input Statistics** The area of optimizing filter banks based on input statistics is an active one. In particular, principal-component filter banks have been shown to be optimal for a wide variety of problems (we

give pointers to the literature on the subject in *Further Reading*). For example, in parallel to our discussion in Chapter 5 on the use of KLT, it is known that the coding gain is maximized if the channel sequences are decorrelated, ( $\tilde{\Phi}_a$  is diagonal), and  $A_\alpha(e^{j\omega}) \geq A_\beta(e^{j\omega})$  if  $\text{var}(\alpha_n) \geq \text{var}(\beta_n)$ . We can diagonalize  $\tilde{\Phi}_a$  by factoring it as  $\tilde{\Phi}_a = Q A Q^*$ , where  $Q$  is the matrix of eigenvectors of  $\tilde{\Phi}_a$ .

## 7.7 Computational Aspects

The power of filter banks is that they are a computational tool; they implement a wide variety of bases (and frames, see Chapter 10). As the two-channel filter bank is the basic building block for many of these, we now spend some time discussing various computational concerns that arise in applications.

### 7.7.1 Two-Channel Filter Banks

We start with a two-channel filter bank with synthesis filters  $\{g, h\}$  and analysis filters  $\{\tilde{g}, \tilde{h}\}$ . For simplicity and comparison purposes, we assume that the input is of even length  $M$ , filters are of even length  $L$ , and all costs are computed per input sample. From (7.80b), the channel signals  $\alpha$  and  $\beta$  are

$$\alpha = \tilde{g}_0 * x_0 + \tilde{g}_1 * x_1, \quad (7.88a)$$

$$\beta = \tilde{h}_0 * x_0 + \tilde{h}_1 * x_1, \quad (7.88b)$$

where  $\tilde{g}_{0,1}, \tilde{h}_{0,1}$  are the polyphase components of the analysis filters  $\tilde{g}$  and  $\tilde{h}$ . We have immediately written the expression in polyphase domain, as it is implicitly clear that it does not make sense to do the filtering first and then discard every other product (see Section 2.9.3).

In general, (7.88) amounts to four convolutions with polyphase components  $x_0$  and  $x_1$ , each of half the original length, plus the necessary additions. Instead of using (2.266), we compute directly the cost per input sample. The four convolutions operate at half the input rate and thus, for every two input samples, we compute  $4L/2$  multiplications and  $4((L/2) - 1) + 2$  additions. This leads to  $L$  multiplications and  $L - 1$  additions/input sample, that is, exactly the same complexity as a convolution by a single filter of size  $L$ . The cost is thus

$$C_{\text{biorth,time}} = 2L - 1 \quad \sim \quad O(L), \quad (7.89)$$

per input sample.

If an FFT-based convolution algorithm is used, for example, overlap-add, we need four convolutions using DFTs of length  $N$  as in (2.265), plus  $2N$  additions. Assume for simplicity and comparison purposes that  $M = L = N2$ .

$$C_{\text{biorth,freq}} = 16\alpha \log_2 L + 14 \quad \sim \quad O(\log_2 L), \quad (7.90)$$

per input sample.



In [121], a precise analysis is made involving FFTs with optimized lengths so as to minimize the operation count. Using the split-radix FFT algorithm, the number of operations becomes (for large  $L$ )

$$C_{\text{biorth,freq,optim}} = 4 \log_2 L + O(\log_2 \log_2 L),$$

again per input sample. Comparing this to  $C_{\text{biorth,freq}}$  (and disregarding the constant  $\alpha$ ), the algorithm starts to be effective for  $L = 8$  and a length-16 FFT, where it achieves around 5 multiplications per sample rather than 8, and leads to improvements of an order of magnitude for large filters (such as  $L = 64$  or  $128$ ). For medium-size filters ( $L = 6, \dots, 12$ ), a method based on fast running convolution is best (see [121]).

Let us now consider some special cases where additional savings are possible.

**Linear-Phase Filter Banks** It is well-known that if a filter is symmetric or anti-symmetric, the number of operations can be halved in (7.89) by simply adding (or subtracting) the two input samples that are multiplied by the same coefficient. This trick can be used in the downsampled case as well, that is, filter banks with linear-phase filters require half the number of multiplications, or  $L/2$  multiplications per input sample (the number of additions remains unchanged), for a total cost of

$$C_{\text{lp,direct}} = \frac{3}{2}L - 1 \quad \sim \quad O(L), \quad (7.91)$$

still  $O(L)$  but with a savings of roughly 25% over (7.89). If the filter length is odd, the polyphase components are themselves symmetric or antisymmetric, and the saving is obvious in (7.88).

Another option is to use a linear-phase lattice factorization:

$$\Phi_p(z) = \alpha \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \prod_{i=1}^{N/2-1} \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} \begin{bmatrix} 1 & \alpha_i \\ \alpha_i & 1 \end{bmatrix}.$$

The individual  $2 \times 2$  symmetric matrices can be written as (we assume  $\alpha_i \neq 1$ )

$$\begin{bmatrix} 1 & \alpha_i \\ \alpha_i & 1 \end{bmatrix} = \frac{1 - \alpha_i}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} \frac{1+\alpha_i}{1-\alpha_i} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

By gathering the scale factors together, we see that each new block in the cascade structure (which increases the length of the filters by two) adds only one multiplication. Thus, we need  $L/4$  multiplications, and  $(L - 1)$  additions per input sample, for a total cost of

$$C_{\text{lp,lattice}} = \frac{5}{4}L - 1 \quad \sim \quad O(L), \quad (7.92)$$

per input sample. The savings is roughly 16% over (7.91), and 37.5% over (7.89).

Two-Channel Filter Bank	$\mu$	$\nu$	Cost	Order
Biorthogonal				
Frequency	$16\alpha \log_2 L$	14	$16\alpha \log_2 L + 14$	$O(\log_2 L)$
Time	$L$	$L - 1$	$2L - 1$	$O(L)$
Linear phase				
Direct form	$(1/2)L$	$L - 1$	$(3/2)L - 1$	$O(L)$
Lattice form	$(1/4)L$	$L - 1$	$(5/4)L - 1$	$O(L)$
Orthogonal				
Lattice form	$(3/4)L$	$(3/4)L$	$(3/2)L$	$O(L)$
Denormalized lattice	$(1/2)L + 1$	$(3/4)L$	$(5/4)L + 1$	$O(L)$
QMF	$(1/2)L$	$(1/2)L$	$L$	$O(L)$

**Table 7.6:** Cost per input sample of computing various two-channel filter banks with length- $L$  filters.

**Orthogonal Filter Banks** As we have seen, there exists a general form for a two-channel paraunitary matrix, given in (7.39). If  $G_0(z)$  and  $G_1(z)$  were of degree zero, it is clear that the matrix in (7.39) would be a rotation matrix, which can be implemented with three multiplications, as we will show shortly. It turns out that for arbitrary-degree polyphase components, terms can still be gathered into rotations, saving 25% of multiplications (at the cost of 25% more additions). This rotation property is more obvious in the lattice structure form of orthogonal filter banks (7.54), where matrices  $R_k$  can be written as:

$$\begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta_i - \sin \theta_i & 0 & 0 \\ 0 & \cos \theta_i + \sin \theta_i & 0 \\ 0 & 0 & \sin \theta_i \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & -1 \end{bmatrix}.$$

Thus, only three multiplications are needed, or  $3L/2$  for the whole lattice. Since the lattice works in the downsampled domain, the cost is  $3L/4$  multiplications per input sample and a similar number of additions, for a total cost of

$$C_{\text{orth,lattice}} = \frac{3}{2}L \sim O(L), \quad (7.93)$$

per input sample. We could also denormalize the diagonal matrix in the above equation (taking out  $\sin \theta_i$  for example) and gather all scale factors at the end of the lattice, leading to  $(L/2 + 1)$  multiplications per input sample, and the same number of additions as before, for a total cost of

$$C_{\text{orth,lattice,denorm}} = \frac{5}{4}L + 1 \sim O(L), \quad (7.94)$$

per input sample.

**QMF Filter Banks** The classic QMF solution discussed in Exercise 7.19, besides using even-length linear phase filters, forces the highpass filter to be equal to the lowpass, modulated by  $(-1)^n$ . The polyphase matrix is therefore:

$$\Phi_p(z) = \begin{bmatrix} G_0(z) & G_1(z) \\ G_0(z) & -G_1(z) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} G_0(z) & 0 \\ 0 & G_1(z) \end{bmatrix},$$

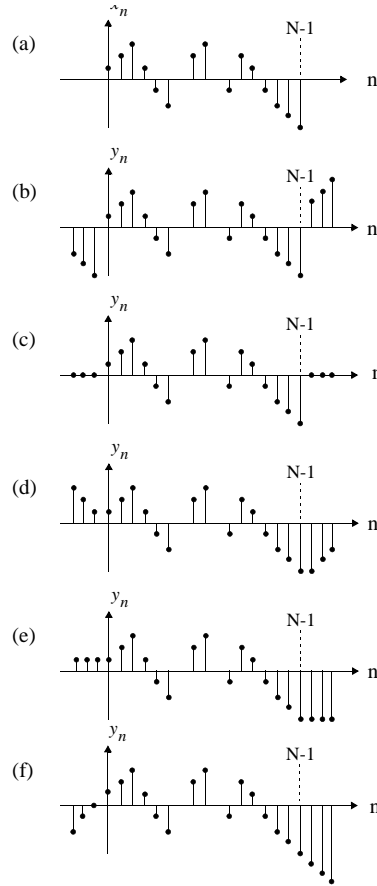
where  $G_0$  and  $G_1$  are the polyphase components of  $G(z)$ . The factorized form on the right indicates that the cost is halved. However, this scheme only approximates a basis expansion (perfect reconstruction) when using FIR filters. Table 7.6 summarizes the costs of various filter banks we have seen so far.

**Multidimensional Filter Banks** While we have not discussed multidimensional filter banks so far (some pointers are given in *Further Reading*), we do touch upon the cost of computing them. For example, filtering an  $M \times M$  image with a filter of length  $L \times L$  requires of the order of  $O(M^2 L^2)$  operations. If the filter is separable, that is,  $G(z_1, z_2) = G_1(z_1)G_2(z_2)$ , then filtering on rows and columns can be done separately and the cost is reduced to an order  $O(2M^2 L)$  operations ( $M$  row filterings and  $M$  column filterings, each using  $ML$  operations).

A multidimensional filter bank can be implemented in its polyphase form, bringing the cost down to the order of a single nondownsampling convolution, just as in the one-dimensional case. A few cases of particular interest allow further reductions in cost. For example, when both filters and downsampling are separable, the system is the direct product of one-dimensional systems, and the implementation is done separately over each dimension. Consider a two-dimensional system filtering an  $M \times M$  image into four subbands using the filters  $\{G(z_1)G(z_2), G(z_1)H(z_2), H(z_1)G(z_2), H(z_1)H(z_2)\}$  each of length  $M \times M$  followed by separable downsampling by two in each dimension. This requires  $M$  decompositions in one dimension (one for each row), followed by  $M$  decompositions in the other, for a total of  $O(2M^2 L)$  multiplications and a similar number of additions. This is a saving of the order of  $L/2$  with respect to the nonseparable case.

### 7.7.2 Boundary Extensions

While most of the literature as well as our exposition implicitly assume infinite-length sequences, in practice this is not the case. Given an  $N \times N$  image, for example, the result of processing it should be another image of the same size. In Chapter 2, we discussed the finite-length case by introducing periodic (circular) extension, when the appropriate convolution is the circular convolution and the appropriate Fourier transform is the DFT. In practice, however, periodic extension is rather artificial as it wraps the sequence around (for example, what is on the left boundary of the image would appear on the right boundary). Other extension are possible, and while for some of them (for example, symmetric), appropriate notions of convolution and Fourier transform are available, in practice this is not done. Instead, different types of extensions are applied (zero-padding, symmetric, continuous, smooth) while still using the tools developed for the periodic extension.



**Figure 7.14:** Boundary extensions. (a) Original sequence  $x$  of length  $N$ . (b) Periodic extension:  $x$  is repeated with a period  $N$ . (c) Zero-padding extension: Beyond the support,  $y$  is set to zero. (d) Symmetric extension: The sequence is flipped at the boundaries to preserve continuity. (Half-point symmetry is shown.) (e) Continuous extension: The boundary value is replicated. (f) Smooth extension: At the boundary, a polynomial extension is applied to preserve higher-order continuity.

Throughout this subsection, we assume a sequence of length  $N$ ; also, we will be using the extension nomenclature adopted in Matlab, and will point out other names under which these extensions are known.

**Periodic Extension** From  $x$ , create a periodic  $y$  as

$$y_n = x_{n \bmod N}.$$

Of those we consider here, this is the only mathematically correct extension in conjunction with the DFT. Moreover, it is simple and works for any sequence

length. The drawback is that the underlying sequence is most likely not periodic, and thus, periodization creates artificial discontinuities at multiples of  $N$ ; see Figure 7.14(b).<sup>113</sup>

**Zero-Padding Extension** From  $x$ , create  $y$  as

$$y_n = \begin{cases} x_n, & n = 0, 1, \dots, N-1; \\ 0, & \text{otherwise.} \end{cases}$$

Again, this extension is simple and works for any sequence length. However, it too creates artificial discontinuities as in Figure 7.14(c). Also, during the filtering process, the sequence is extended by the length of the filter (minus 1), which is often undesirable.

**Symmetric Extension** From  $x$ , create a double-length  $y$  as

$$y_n = \begin{cases} x_n, & n = 0, 1, \dots, N-1; \\ x_{2N-n-1}, & n = N, N+1, \dots, 2N-1, \end{cases}$$

and then periodize it. As shown in Figure 7.14(d), this periodic sequence of period  $2N$  does not show the artificial discontinuities of the previous two cases.<sup>114</sup> However, the sequence is now twice as long, and unless carefully treated, this redundancy is hard to undo. Cases where it can be handled easily are when the filters are symmetric or antisymmetric, because the output of the filtering will be symmetric or antisymmetric as well.

There exist two versions of the symmetric extension, depending on whether whole- or half-point symmetry is used. The formulation above is called *half-point symmetric* because  $y$  is symmetric about the half-integer index value  $N - \frac{1}{2}$ . An alternative is *whole-point symmetric*  $y$

$$y_n = \begin{cases} x_n, & n = 0, 1, \dots, N-1; \\ x_{2N-n-2}, & n = N, N+1, \dots, 2N-2, \end{cases}$$

with even symmetry around  $N$ .

**Continuous Extension** From  $x$ , create a double-length  $y$  as

$$y_n = \begin{cases} x_n, & n = 0, 1, \dots, N-1; \\ x_{N-1}, & n = N, N+1, \dots, 2N-1; \\ x_0, & n = 0, -1, \dots, -N+1, \end{cases}$$

shown in Figure 7.14(e). This extension is also called *boundary replication extension*. It is a relatively smooth extension and is often used in practice.

<sup>113</sup>Technically speaking, a discrete sequence cannot be continuous or discontinuous. However, if the sequence is a densely sampled version of a smooth sequence, periodization will destroy this smoothness.

<sup>114</sup>It does remain discontinuous in its derivatives however; for example, if it is linear, it will be smooth but not differentiable at 0 and  $N$ .

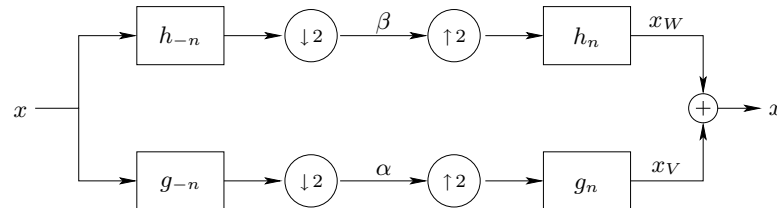
**Smooth Extension** Another idea is to extend the sequence by polynomial extrapolation, as in Figure 7.14(f). This is only lightly motivated at this point, but after we establish polynomial approximation properties of the discrete wavelet transforms in Chapter 9, it will be clear that a sequence extension by polynomial extrapolation will be a way to get zeros as detail coefficients. The order of the polynomial is such that on the one hand, it gets annihilated by the zero moments of the wavelet, and on the other hand, it can be extrapolated by the lowpass filter.

## Chapter at a Glance

Our goal in this chapter was to use signal processing machinery to build discrete-time bases with structure in terms of time-frequency localization properties. Moreover, we restricted ourselves to those bases generated by two prototype sequences, one that together with its shifts covers the space of lowpass sequences, and the other that together with its shifts covers the space of highpass sequences. The signal processing tool implementing such bases is a *two-channel filter bank*.

### Two-Channel Filter Bank

#### Block diagram



#### Basic characteristics

number of channels	$M = 2$
sampling factor	$N = 2$
channel sequences	$\alpha_n \quad \beta_n$

#### Filters

#### Synthesis

#### Analysis

	lowpass	highpass	lowpass	highpass
orthogonal	$g_n$	$h_n$	$g_{-n}$	$h_{-n}$
biorthogonal	$g_n$	$h_n$	$\tilde{g}_n$	$\tilde{h}_n$
polyphase components	$g_{0,n}, g_{1,n}$	$h_{0,n}, h_{1,n}$	$\tilde{g}_{0,n}, \tilde{g}_{1,n}$	$\tilde{h}_{0,n}, \tilde{h}_{1,n}$

**Table 7.7:** Two-channel filter bank.

Haar Filter Bank	Synthesis		Analysis	
	lowpass	highpass	lowpass	highpass
Time domain	$g_n$ $(\delta_n + \delta_{n-1})/\sqrt{2}$	$h_n$ $(\delta_n - \delta_{n-1})/\sqrt{2}$	$g_{-n}$ $(\delta_n + \delta_{n+1})/\sqrt{2}$	$h_{-n}$ $(\delta_n - \delta_{n+1})/\sqrt{2}$
$z$ -domain	$G(z)$ $(1 + z^{-1})/\sqrt{2}$	$H(z)$ $(1 - z^{-1})/\sqrt{2}$	$G(z^{-1})$ $(1 + z)/\sqrt{2}$	$H(z^{-1})$ $(1 - z)/\sqrt{2}$
DTFT domain	$G(e^{j\omega})$ $(1 + e^{-j\omega})/\sqrt{2}$	$H(e^{j\omega})$ $(1 - e^{-j\omega})/\sqrt{2}$	$G(e^{-j\omega})$ $(1 + e^{j\omega})/\sqrt{2}$	$H(e^{-j\omega})$ $(1 - e^{j\omega})/\sqrt{2}$

**Table 7.8:** Haar filter bank in various domains.

**Two-Channel Orthogonal Filter Bank****Relationship between lowpass and highpass filters**

Time domain	$\langle h_n, g_{n-2k} \rangle_n = 0$
Matrix domain	$D_2 H^T G U_2 = 0$
$z$ domain	$G(z)H(z^{-1}) + G(-z)H(-z^{-1}) = 0$
DTFT domain	$G(e^{j\omega})H(e^{j\omega}) + G(e^{j(\omega+\pi)})H(e^{j(\omega+\pi)}) = 0$
Polyphase domain	$G_0(z)G_1(z^{-1}) + H_0(z)H_1(z^{-1}) = 0$

**Sequences****Basis**

Frequency domain	lowpass	highpass
Time domain	$\{g_{n-2k}\}_{k \in \mathbb{Z}}$	$\{h_{n-2k}\}_{k \in \mathbb{Z}}$

**Filters****Synthesis****Analysis**

	lowpass	highpass	lowpass	highpass
Time domain	$g_n$	$\pm(-1)^n g_{-n+2\ell-1}$	$g_{-n}$	$\pm(-1)^n g_{n+2\ell-1}$
$z$ domain	$G(z)$	$\mp z^{-2\ell+1} G(-z^{-1})$	$G(z^{-1})$	$\mp z^{2\ell-1} G(-z)$
DTFT domain	$G(e^{j\omega})$	$\mp e^{j(-2\ell+1)\omega} G(e^{-j(\omega+\pi)})$	$G(e^{-j\omega})$	$\mp e^{j(2\ell-1)\omega} G(e^{j(\omega+\pi)})$

**Matrix view****Basis**

Time domain	$\Phi$	$\begin{bmatrix} \dots & g_{n-2k} & h_{n-2k} & \dots \end{bmatrix}$
$z$ domain	$\Phi(z)$	$\begin{bmatrix} G(z) & H(z) \\ G(-z) & H(-z) \end{bmatrix}$
DTFT domain	$\Phi(e^{j\omega})$	$\begin{bmatrix} G(e^{j\omega}) & H(e^{j\omega}) \\ G(e^{j(\omega+\pi)}) & H(e^{j(\omega+\pi)}) \end{bmatrix}$
Polyphase domain	$\Phi_p(z)$	$\begin{bmatrix} G_0(z) & H_0(z) \\ G_1(z) & H_1(z) \end{bmatrix}$

**Constraints****Orthogonality relations****Perfect reconstruction**

Time domain	$\Phi^T \Phi = I$	$\Phi \Phi^T = I$
$z$ domain	$\Phi(z^{-1})^T \Phi(z) = I$	$\Phi(z) \Phi^T(z^{-1}) = I$
DTFT domain	$\Phi^T(e^{-j\omega}) \Phi(e^{j\omega}) = I$	$\Phi(e^{j\omega}) \Phi^T(e^{-j\omega}) = I$
Polyphase domain	$\Phi_p^T(z^{-1}) \Phi_p(z) = I$	$\Phi_p(z) \Phi_p^T(z^{-1}) = I$

**Table 7.9:** Properties of an orthogonal two-channel filter bank.



**Two-Channel Biorthogonal Filter Bank****Relationship between lowpass and highpass filters**

Time domain	$\langle h_n, g_{n-2k} \rangle_n = 0$
Matrix domain	$D_2 H^T G U_2 = 0$
$z$ domain	$G(z)H(z^{-1}) + G(-z)H(-z^{-1}) = 0$
DTFT domain	$G(e^{j\omega})H(e^{-j\omega}) + G(e^{j(\omega+\pi)})H(e^{j(-\omega+\pi)}) = 0$
Polyphase domain	$G_0(z)G_1(z^{-1}) + H_0(z)H_1(z^{-1}) = 0$

**Sequences****Basis**

lowpass	highpass
$\{g_{n-2k}\}_{k \in \mathbb{Z}}$	$\{h_{n-2k}\}_{k \in \mathbb{Z}}$

**Dual basis**

lowpass	highpass
$\{\tilde{g}_{2k-n}\}_{k \in \mathbb{Z}}$	$\{\tilde{h}_{2k-n}\}_{k \in \mathbb{Z}}$

**Filters****Synthesis**

	lowpass	highpass
Time domain	$g_n$	$\pm(-1)^n \tilde{g}_{n-2\ell+1}$
$z$ domain	$G(z)$	$\mp z^{-2\ell+1} \tilde{G}(-z)$
DTFT domain	$G(e^{j\omega})$	$\mp e^{j(-2\ell+1)\omega} \tilde{G}(e^{j(\omega+\pi)})$

**Analysis**

	lowpass	highpass
	$\tilde{g}_n$	$\pm(-1)^n g_{n+2\ell-1}$
	$\tilde{G}(z)$	$\mp z^{2\ell-1} G(-z)$
	$\tilde{G}(e^{j\omega})$	$\mp e^{j(2\ell-1)\omega} G(e^{j(\omega+\pi)})$

**Matrix view****Basis**

Time domain	$\Phi$	$\begin{bmatrix} \dots & g_{n-2k} & h_{n-2k} & \dots \end{bmatrix}$
$z$ domain	$\Phi(z)$	$\begin{bmatrix} G(z) & H(z) \\ G(-z) & H(-z) \end{bmatrix}$
DTFT domain	$\Phi(e^{j\omega})$	$\begin{bmatrix} G(e^{j\omega}) & H(e^{j\omega}) \\ G(e^{j(\omega+\pi)}) & H(e^{j(\omega+\pi)}) \end{bmatrix}$
Polyphase domain	$\Phi_p(z)$	$\begin{bmatrix} G_0(z) & H_0(z) \\ G_1(z) & H_1(z) \end{bmatrix}$

**Dual basis**

	$\tilde{\Phi}$	$\begin{bmatrix} \dots & \tilde{g}_{2k-n} & \tilde{h}_{2k-n} & \dots \end{bmatrix}$
	$\tilde{\Phi}(z)$	$\begin{bmatrix} \tilde{G}(z) & \tilde{H}(z) \\ \tilde{G}(-z) & \tilde{H}(-z) \end{bmatrix}$
	$\tilde{\Phi}(e^{j\omega})$	$\begin{bmatrix} \tilde{G}(e^{j\omega}) & \tilde{H}(e^{j\omega}) \\ \tilde{G}(e^{j(\omega+\pi)}) & \tilde{H}(e^{j(\omega+\pi)}) \end{bmatrix}$
	$\tilde{\Phi}_p(z)$	$\begin{bmatrix} \tilde{G}_0(z) & \tilde{H}_0(z) \\ \tilde{G}_1(z) & \tilde{H}_1(z) \end{bmatrix}$

**Constraints****Biorthogonality relations**

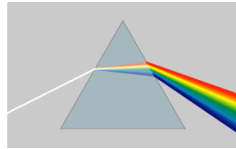
Time domain	$\Phi^T \tilde{\Phi} = I$
$z$ domain	$\Phi(z)^T \tilde{\Phi}(z) = I$
DTFT domain	$\Phi^T(e^{j\omega}) \tilde{\Phi}(e^{j\omega}) = I$
Polyphase domain	$\Phi_p^T(z) \tilde{\Phi}_p(z) = I$

**Perfect reconstruction**

	$\Phi \tilde{\Phi}^T = I$
	$\Phi(z) \tilde{\Phi}^T(z) = I$
	$\Phi(e^{j\omega}) \tilde{\Phi}^T(e^{j\omega}) = I$
	$\Phi_p(z) \tilde{\Phi}_p^T(z) = I$

**Table 7.10:** Properties of a biorthogonal two-channel filter bank.

## Historical Remarks



Filter banks have been popular in signal processing since the 1970s when the question of critically-sampled filter banks, those with the number of channel samples per unit of time conserved, arose in the context of subband coding of speech. In that method, a speech sequence is split into downsampled frequency bands, allowing for more powerful compression. However, downsampling can create a perceptually disturbing effect known as aliasing, prompting Esteban and Galand [52] in 1977 to propose a simple and elegant *quadrature mirror filters (QMF)* aliasing-removal technique. As QMF solution does not allow for perfect reconstruction, a flurry of work followed to solve the problem. Mintzer [105] as well Smith and Barnwell [135] proposed an orthogonal solution independently in the mid 1980s. Vaidyanathan [156] established a connection to lossless systems, unveiling the factorization and design of paraunitary matrices [160]. For wavelet purposes, Daubechies then designed filters with a maximum number of zeros at  $z = -1$  [39], a solution that goes back to Herrmann's design of maximally flat FIR filters [74]. The equivalent IIR filter design problem leads to Butterworth filters, as derived by Herley and Vetterli [72]. Vetterli solved the biorthogonal filter bank problem [164, 165], while Cohen, Daubechies and Feauveau [30] as well as Vetterli and Herley [166] tackled those with maximum number of zeros at  $z = -1$ . The poly-phase framework was used by many authors working on filter banks, but really goes back to earlier work on transmultiplexers by Bellanger and Daguet [8]. The realization that perfect reconstruction subband coding can be used for perfect transmultiplexing appears in [165]. The idea of multichannel structures that can be inverted perfectly, including with quantization, goes back to ladder structures in filter design and implementation, in the works of Bruckens and van den Enden, Marshall, Shah and Kalker [20, 103, 129]. Sweldens generalized this idea under the name of lifting [147], deriving a number of new schemes based on this concept, including filter banks with nonlinear operators and nonuniform sampling.

## Further Reading

**Books and Textbooks** A few standard textbooks on filter banks exist, written by Vaidyanathan [158], Vetterli and Kovačević [167], Strang and Nguyen [143], among others.

**$N$ -Channel Filter Banks** One of the important and immediate generalizations of two-channel filter banks is when we allow the number of channels to be  $N$ . Numerous options are available, from directly designing  $N$ -channel filter banks, studied in detail by Vaidyanathan [156, 157], through those built by cascading filter banks with different number of branches, leading to almost arbitrary frequency divisions. The analysis methods follow closely those of the two-channel filter banks, albeit with more freedom; for example, orthogonality and linear phase are much easier to achieve at the same time. We discuss  $N$ -channel filter banks in detail in Chapter 8, with special emphasis on local Fourier bases.

**Multidimensional Filter Banks** The first difference we encounter when dealing with multidimensional filter banks is that of sampling. Regular sampling with a given density can be accomplished using any number of sampling lattices, each having any number of

associated sampling matrices. These have been described in detail by Dubois in [47], and have been used by Viscito and Allebach [169], Karlsson and Vetterli [84], Kovačević and Vetterli [91], Do and Vetterli [43], among others, to design multidimensional filter banks. Apart from the freedom coming with different sampling schemes, the associated filters can now be truly multidimensional, allowing for a much larger space of solutions.

**IIR Filter Banks** While IIR filters should be of importance because of their good frequency selectivity and computational efficiency, they have not been used extensively as their implementation in a filter-bank framework comes at a cost: one side of the filter bank is necessarily anticausal. They have found some use in image processing as the finite length of the input allows for storing the state in the middle of the filter bank and synthesizing from that stored state. Coverage of IIR filter banks can be found in [118, 134, 72].

**Oversampled Filter Banks** Yet another generalization occurs when we allow for redundancy, leading to overcomplete filter banks implementing frame expansions, covered in Chapter 10. These filter banks are becoming popular in applications due to inherent freedom in design.

**Complex-Coefficient Filter Banks** This entire chapter dealt exclusively with real-coefficient filter banks, due to their prevalence in practice. Complex-coefficient filter banks exist, from the very early QMFs [107] to more recent ones, mostly in the form of complex exponential-modulated local Fourier bases, discussed in Section 8.3, as well as the redundant ones, such as Gabor frames [36, 16, 15, 14, 53], discussed in Chapter 10.

**QMF Filter Banks** QMF filter banks showed the true potential of filter banks, as it was clear that one could have nonideal filters and still split and reconstruct the input spectrum. The excitement was further spurred by the famous linear-phase designs by Johnston [81] in 1980. Exercise 7.19 discusses derivation of these filters and their properties.

**Time-Varying Filter Banks and Boundary Filters** The periodic shift variance of filter banks can be exploited to change a filter bank essentially every period. This was done for years in audio coding through the so-called MDCT filter banks, discussed in Section 8.4.1. Herley and Vetterli proposed a more formal approach in [73], by designing different filters to be used at the boundary of a finite-length input, or a filter-bank change.

**Transmultiplexing** The dual scheme to a filter bank is known as a transmultiplexer, where two sequences are synthesized into a combined sequence from which the two parts can be extracted perfectly. An orthogonal decomposition with many channels leads to *orthogonal frequency-division multiplexing (OFDM)*, the basis for many modulation schemes used in communications, such as 802.11. The analysis of transmultiplexers uses similar tools as for filter banks [165], covered in Solved Exercise 7.7 for the orthogonal case, and in Exercise 7.20 for the biorthogonal case. Exercise 7.21 considers frequency-division multiplexing with Haar filters.

**Filter Banks with Stochastic Inputs** Among the wealth of filter banks available, it is often necessary to determine which one is the most suitable for a given application. A number of measures have been proposed, for example, quantifying shift variance of subband energies for deterministic inputs. Similarly, in [1, 2], the author proposes, among others, a counterpart measure based on the cyclostationarity of subband powers. We do not dwell on

these here, rather we leave the discussion for Chapter 13. Akkarakaran and Vaidyanathan in [4] discuss bifrequency and bispectrum maps (deterministic and stochastic time-varying autocorrelations) in filter banks and answer many relevant questions; some similar issues are tackled by Therrien in [148]. In [159], Vaidyanathan and Akkarakaran give a review of optimal filter banks based on input statistics. In particular, principal-component filter banks offer optimal solutions to various problems, some of these discussed in [151] and [152, 153].

## Exercises with Solutions

### 7.1. Lowpass Projection in $z$ -Domain

In (7.18), the projection property of the lowpass channel in an orthogonal two-channel filter bank was shown using operator notation. An alternative is to use  $z$ -transforms and derive  $X_V(z)$  as the output of the lowpass channel.

- (i) Show that the transpose of the convolution operator corresponding to the filter with  $z$ -transform  $G(z)$  is a convolution operator with the filter  $G(z^{-1})$ .
- (ii) Using the  $z$ -domain expression for downsampling followed by upsampling, (2.193), derive  $X_V(z)$ ; since it is the  $z$ -domain expression for  $x_V$ , it is self-adjoint as we have proved in the text.
- (iii) Using the above, confirm that if  $X_V(z)$  is used as input to the lowpass channel, its output is identical, verifying idempotency.

*Solution:*

- (i) We use the matrix notation for the convolution operator  $G$  in (2.63) and its corresponding adjoint,  $G^*$  in (2.65). It is clear that the adjoint is just the time-reversed version of  $G$ . According to (2.137), the  $z$ -transform of the adjoint of  $G$  is then  $G(z^{-1})$ .
- (ii) Using (2.193) on  $G(z^{-1})X(z)$ ,

$$X_V(z) = \frac{1}{2}G(z)(G(z^{-1})X(z) + G(-z^{-1})X(-z)). \quad (\text{E7.1-1})$$

- (iii) We now show idempotency,

$$\begin{aligned} & \frac{1}{2}G(z)[G(z^{-1})X_V(z) + G(-z^{-1})X_V(-z)] \\ &= \frac{1}{2}G(z)[G(z^{-1})(\frac{1}{2}G(z)(G(z^{-1})X(z) + G(-z^{-1})X(-z))) + \\ & \quad G(-z^{-1})(\frac{1}{2}G(-z)(G(-z^{-1})X(-z) + G(z^{-1})X(z)))] \\ &= \frac{1}{2}G(z)[G(z)G(z^{-1})G(z^{-1})X(z) + G(z)G(z^{-1})G(-z^{-1})X(-z) + \\ & \quad G(-z)G(-z^{-1})G(-z^{-1})X(-z) + G(-z^{-1})G(-z)G(z^{-1})X(z)] \\ &= \frac{1}{2}G(z)\frac{1}{2}[G(z)G(z^{-1}) + G(-z)G(-z^{-1})][G(z^{-1})X(z) + G(-z^{-1})X(-z)] \\ &\stackrel{(a)}{=} \frac{1}{2}G(z)(G(z^{-1})X(z) + G(-z^{-1})X(-z)) \stackrel{(b)}{=} X_V(z), \end{aligned}$$

where (a) follows from the orthogonality of the lowpass filter, (7.13); and (b) from the expression for  $X_V(z)$  we just derived, (E7.1-1).

### 7.2. Even-Length Requirement for Orthogonality

Given is an orthogonal FIR filter with impulse response  $g$  satisfying

$$\langle g_n, g_{n-2k} \rangle_n = \delta_k.$$

Prove that its length  $L$  is necessarily an even integer.

*Solution:* We are given an orthogonal, causal FIR filter  $g$  of length  $L$ . This implies that  $g_0 \neq 0$ ,  $g_{L-1} \neq 0$ . The orthogonality of the filter can be expressed through its deterministic

autocorrelation as in (7.15). Since  $a_n = g_n * g_{-n}$  (see (2.61d)), we can write  $A(z)$  as

$$A(z) = \sum_{i=-(L-1)}^{L-1} a_i z^{-i}.$$

From  $A(z) + A(-z) = 2$  we know that all even-indexed coefficients of  $A(z)$  must be zero. Since  $a_{L-1} = a_{-L+1} = g_0 g_{L-1}$ , and both  $g_0 \neq 0$  and  $g_{L-1} \neq 0$ , then  $a_{L-1} = a_{-L+1} \neq 0$ , which means that this is an odd-indexed coefficient, or,  $L$  is even.

### 7.3. Lattice Factorization Design

Consider the lattice factorization of orthogonal filter banks (7.54).

- (i) If  $U$  is a rotation matrix as in (1.220a), prove that the sum of angles must satisfy (7.58a) for the lowpass filter to have a zero at  $z = -1$  ( $\omega = \pi$ ).
- (ii) Which condition would the angles have to satisfy in (i) if, instead of  $G(z)|_{z=1} = \sqrt{2}$  as in (7.56b), it had been  $G(z)|_{z=1} = -\sqrt{2}$  (simple change of phase)?
- (iii) If  $U$  is a rotoinversion matrix as in (1.220b), prove that the sum of angles must satisfy (7.58b) for the lowpass filter to have a zero at  $z = -1$  ( $\omega = \pi$ ).
- (iv) If  $U$  is a rotation matrix and  $K = 2$  (leading to length-4 orthogonal filters), impose two zeros at  $z = -1$  and show that a possible solution is  $\theta_0 = \pi/3$  and  $\theta_1 = -\pi/12$  (the same solution as in Example 7.3, one of the filters from the Daubechies family).

*Solution:* We use the lattice factorization of orthogonal filter banks (7.54).

- (i) To solve this part, we must solve (7.57):

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \underbrace{\begin{bmatrix} G_0(1) \\ G_0(1) \end{bmatrix}}_G = \begin{bmatrix} \sqrt{2} \\ 0 \end{bmatrix}.$$

As  $G$  is just the first column of  $\Phi_p(z^2)|_{z=-1}$ , we compute it to find:

$$\Phi_p(z^2)|_{z=-1} = \left( R_0 \prod_{k=1}^{K-1} \begin{bmatrix} 1 & 0 \\ 0 & z^{-2} \end{bmatrix} R_k \right) \Big|_{z=-1} = R_0 \prod_{k=1}^{K-1} R_k = R,$$

where  $R$  is a rotation matrix with angle  $\theta = \sum_{k=0}^{K-1} \theta_k$  (intuitively, this is clear, as a rotation by a succession of angles is equivalent to a rotation by the sum of all angles). We thus rewrite (7.57) as

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} = \begin{bmatrix} \sqrt{2} \\ 0 \end{bmatrix} \quad \Leftrightarrow \quad \begin{aligned} \cos \theta + \sin \theta &= \sqrt{2}, \\ \cos \theta - \sin \theta &= 0, \end{aligned} \quad (\text{E7.3-1})$$

the solution of which is (7.58a).

- (ii) We repeat the above process, except that the set of equations we must solve is

$$\begin{aligned} \cos \theta + \sin \theta &= -\sqrt{2}, \\ \cos \theta - \sin \theta &= 0, \end{aligned}$$

the solution of which is (7.58b).

- (iii) If  $U$  is a rotoinversion matrix, the total angle is  $\theta = \theta_0 - \sum_{k=1}^{K-1} \theta_k$  since

$$\begin{bmatrix} \cos \theta_0 & \sin \theta_0 \\ \sin \theta_0 & -\cos \theta_0 \end{bmatrix} \begin{bmatrix} \cos \theta_1 & -\sin \theta_1 \\ \sin \theta_1 & \cos \theta_1 \end{bmatrix} = \begin{bmatrix} \cos(\theta_0 - \theta_1) & \sin(\theta_0 - \theta_1) \\ \sin(\theta_0 - \theta_1) & -\cos(\theta_0 - \theta_1) \end{bmatrix},$$

that is, a rotation followed by rotoinversion is a rotoinversion again. Solving (E7.3-1) gives the desired result.

- (iv) We now want to impose two zeros at  $z = -1$ . We compute

$$\begin{aligned} \Phi_p(z) &= R_0 \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} R_1 \\ &= \begin{bmatrix} \cos \theta_0 \cos \theta_1 - \sin \theta_0 \sin \theta_1 z^{-1} & -\cos \theta_0 \sin \theta_1 - \sin \theta_0 \cos \theta_1 z^{-1} \\ \sin \theta_0 \cos \theta_1 + \cos \theta_0 \sin \theta_1 z^{-1} & -\sin \theta_0 \sin \theta_1 + \cos \theta_0 \cos \theta_1 z^{-1} \end{bmatrix}. \end{aligned}$$

## 622 Chapter 7. Filter Banks: Building Blocks of Time-Frequency Expansions

Using (7.32) and (7.33), we find

$$G(z) = \cos \theta_0 \cos \theta_1 + \sin \theta_0 \cos \theta_1 z^{-1} - \sin \theta_0 \sin \theta_1 z^{-2} + \cos \theta_0 \sin \theta_1 z^{-3}.$$

The first zero at  $z = -1$  we have already solved for; we know that  $\theta_0 + \theta_1 = \pi/4$ . We now solve for an additional zero by differentiating in the Fourier domain:

$$\left. \frac{\partial G(e^{j\omega})}{\partial \omega} \right|_{\omega=\pi} = \sin \theta_0 \cos \theta_1 + 2 \sin \theta_0 \sin \theta_1 + 3 \cos \theta_0 \sin \theta_1 = 0.$$

Using trigonometric identities, we can rewrite this as:

$$\begin{aligned} & \frac{1}{2} (\sin(\theta_0 + \theta_1) + \sin(\theta_0 - \theta_1) - 2 \cos(\theta_0 + \theta_1) + 2 \cos(\theta_0 - \theta_1)) \\ & + 3 \sin(\theta_0 + \theta_1) - 3 \sin(\theta_0 - \theta_1) \\ & = \frac{1}{\sqrt{2}} - \sin(\theta_0 - \theta_1) + \cos(\theta_0 - \theta_1) = 0, \end{aligned}$$

leading to

$$\cos(2\theta_0 - \frac{\pi}{4}) - \sin(2\theta_0 - \frac{\pi}{4}) = -\frac{1}{\sqrt{2}}.$$

Using again trigonometric identities, we get that  $2\theta_0 = 2\pi/3$ , or,  $\theta_0 = \pi/3$  and  $\theta_1 = -\pi/12$ .

### 7.4. Polynomial Reproduction in Perfect Reconstruction Biorthogonal Filter Banks

Given is a biorthogonal filter bank with a synthesis lowpass filter of the form

$$G(z) = (1 + z^{-1})^N R(z). \quad (\text{E7.4-1})$$

Prove that in such a filter bank:

- (i) The analysis highpass filter  $\tilde{H}(z)$  has  $(N - 1)$  zero moments.
- (ii) Polynomials of degree up to  $(N - 1)$  belong to the space  $V = \overline{\text{span}}(\{g_{n-2k}\}_{k \in \mathbb{Z}})$ , that is,  $g$  and its even shifts can reproduce polynomials up to degree  $(N - 1)$ .

*Solution:* This solution builds up on the solution of Exercise 7.6.

- (i) Given the synthesis filter  $G(z)$  from (E7.4-1), from (7.75b), we know that the analysis highpass filter  $\tilde{H}(z)$  can be computed as

$$\tilde{H}(z) = -z^{L-1} G(-z) = -z^{L-1} (1 - z^{-1})^N R(-z) = -z(1 - z^{-1})^N R(-z).$$

where we arbitrarily set  $L = 2$ . Following Exercise 7.6, the above filter has  $(N - 1)$  zero moments.

- (ii) Since  $\tilde{H}$  has  $N$  zeros at  $z = 1$ , the highpass channel of the perfect reconstruction filter bank outputs a zero when presented with a polynomial of degree  $(N - 1)$  as input. Since we have the perfect reconstruction property, we know that the output must then be wholly preserved in the lowpass channel. Thus, polynomials up to degree  $(N - 1)$  belong to the space  $V = \overline{\text{span}}(\{g_{n-2k}\}_{k \in \mathbb{Z}})$ .

### 7.5. Linear Phase Filter Banks

A linear phase two-channel perfect reconstruction filter bank has a 4-tap symmetric synthesis lowpass filter  $G(z)$  with a zero at  $\omega = 2\pi/3$ .

- (i) Within a scaling factor, what is  $G(z)$ ?
- (ii) Prove that the analysis lowpass filter  $\tilde{G}(z)$  cannot be a 2-tap filter.
- (iii) Within a scaling factor, what is  $\tilde{G}(z)$ ? (You may assume that  $H(z)$  is of shortest possible length and that the delay is such so as to make  $\tilde{G}(z)$  causal.)

*Solution:*

- (i) The filter is of the form

$$G(z) = a + bz^{-1} + bz^{-2} + az^{-3} = a(1 + z^{-3}) + b(z^{-1} + z^{-2}).$$

We are given that the filter has a zero at  $\omega = 2\pi/3$ . Thus

$$G(e^{j2\pi/3}) = a(1 + 1) + b(-1/2 - j\sqrt{3}/2 - 1/2 + j\sqrt{3}/2) = 2a - b = 0,$$

leading to  $b = 2a$  and thus

$$G(z) = a(1 + 2z^{-1} + 2z^{-2} + z^{-3}).$$

- (ii) According to Proposition 7.11, since we have an even-length, symmetric filter, the other filter will be antisymmetric and of even length, differing by an even multiple of 2. Thus,  $H(z)$  can be of length 4, 8, 12, ... Using (7.75a) we get that

$$\tilde{G}(z) = z^{-k}H(-z),$$

and thus,  $\tilde{G}(z)$  can only be of length 4, 8, 12, ...

- (iii) Since  $H(z)$  is antisymmetric and of length 4, we get

$$H(z) = c + dz^{-1} - dz^{-2} - cz^{-3} = c(1 - z^{-3}) + d(z^{-1} - z^{-2}).$$

Writing the determinant of the analysis polyphase matrix and forcing it to be a delay

$$(d - 2c)(1 + z^{-2}) + 2(2d - c)z^{-1} = pz^{-l},$$

leading to  $d = 2c$ . Therefore,

$$\tilde{G}(z) = z^{-k}H(-z) = z^{-k}c(1 - 2z^{-1} - 2z^{-2} + z^{-3}),$$

and choosing  $k = 0$  to leave  $\tilde{G}$  causal

$$\tilde{G}(z) = c(1 - 2z^{-1} - 2z^{-2} + z^{-3}).$$

#### 7.6. Filter Design

Let

$$A(z) = \frac{1}{16}(1+z)^2(1+z^{-1})^2(-z+4-z^{-1}).$$

- (i) Find the roots of  $A(z)$ .
- (ii) Based on the factorization of this polynomial, find all the possible nontrivial *orthogonal* filters. Verify that your solutions are power complementary.
- (iii) Based on the factorization of this polynomial, construct all possible nontrivial linear phase (having a center of (anti-)symmetry) *biorthogonal* filters satisfying the constraint  $\tilde{H}(1) = 1$ .
- (iv) Using Matlab, plot the magnitude responses of all of the filters obtained in (ii) and (iii) and compare their shape.

*Solution:*

- (i)  $A(z)$  has 4 roots at  $z = -1$  and 2 roots at  $2 + \sqrt{3}$  and  $2 - \sqrt{3}$ . We can therefore factor  $A(z)$  as

$$A(z) = \frac{1}{16(2 + \sqrt{3})}(1+z)^2(1+z^{-1})^2(1 - (2 + \sqrt{3})z)(1 - (2 + \sqrt{3})z^{-1}).$$

- (ii) For the filters  $G(z)$  to be orthonormal, we need to have  $A(z) = G(z)G(z^{-1})$ . Table E7.6-1 lists possible choices.
- (iii) To have a pair of biorthogonal filters, we need to find filters  $G(z)$  and  $\tilde{G}(z)$  such that  $A(z) = G(z)\tilde{G}(z)$ . Additionally, the filters need to be linear phase and have  $\tilde{G}(1) = 1$ . Table E7.6-2 lists possible choices. Note that each  $\tilde{G}(z)$  can be multiplied by  $(-z + 4 - z^{-1})/2$ , as long as the corresponding  $G(z)$  is divided by the same  $(-z + 4 - z^{-1})/2$ . More generally, each  $\tilde{G}(z)$  can be multiplied by any arbitrary polynomial  $A(z)$ , with  $A(1) = 1$ , as long as the corresponding  $G(z)$  is divided by the same  $A(z)$ . Note that in most cases this will lead to an IIR filter  $G(z)$ .

#### 7.7. Two-Channel Transmultiplexer

A *transmultiplexer* is a two-channel filter bank with a reversed order of analysis and synthesis parts, as shown in Figure E7.7-1. If the inputs into the synthesis filter bank are  $\alpha(z)$  and  $\beta(z)$ , and the output of the synthesis bank is  $X(z)$ , find the input-output relationship of the transmultiplexer and comment.

*Solution:* Given that we know a sequence can be split into two bands and perfectly reconstructed, can we do the converse? Intuitively, it seems that such a scheme should work, and indeed it does, as we show shortly. Beyond the algebraic result, transmultiplexers are of

## 624 Chapter 7. Filter Banks: Building Blocks of Time-Frequency Expansions

$G(z)$
$\frac{\pm 1}{4\sqrt{2+\sqrt{3}}}(1+z)(1+z^{-1})(1-(2+\sqrt{3})z)$
$\frac{\pm 1}{4\sqrt{2+\sqrt{3}}}(1+z)(1+z^{-1})(1-(2+\sqrt{3})z^{-1})$
$\frac{\pm 1}{4\sqrt{2+\sqrt{3}}}(1+z)^2(1-(2+\sqrt{3})z)$
$\frac{\pm 1}{4\sqrt{2+\sqrt{3}}}(1+z)^2(1-(2+\sqrt{3})z^{-1})$
$\frac{\pm 1}{4\sqrt{2+\sqrt{3}}}(1+z^{-1})^2(1-(2+\sqrt{3})z)$
$\frac{\pm 1}{4\sqrt{2+\sqrt{3}}}(1+z^{-1})^2(1-(2+\sqrt{3})z^{-1})$

**Table E7.6-1:** Possible choices for the orthogonal filter  $G(z)$  in Exercise 7.6.

$\tilde{G}(z)$	$G(z)$
1	$\frac{1}{16(2+\sqrt{3})}(1+z)^2(1+z^{-1})^2(1-(2+\sqrt{3})z)(1-(2+\sqrt{3})z^{-1})$
$\frac{1}{2}(1+z)$	$\frac{1}{8\sqrt{2+\sqrt{3}}}(1+z)(1+z^{-1})^2(1-(2+\sqrt{3})z)(1-(2+\sqrt{3})z^{-1})$
$\frac{1}{2}(1+z^{-1})$	$\frac{1}{8\sqrt{2+\sqrt{3}}}(1+z)^2(1+z^{-1})(1-(2+\sqrt{3})z)(1-(2+\sqrt{3})z^{-1})$
$\frac{1}{4}(1+z)^2$	$\frac{1}{4\sqrt{2+\sqrt{3}}}(1+z^{-1})^2(1-(2+\sqrt{3})z)(1-(2+\sqrt{3})z^{-1})$
$\frac{1}{4}(1+z^{-1})^2$	$\frac{1}{4\sqrt{2+\sqrt{3}}}(1+z)^2(1-(2+\sqrt{3})z)(1-(2+\sqrt{3})z^{-1})$
$\frac{1}{4}(1+z)(1+z^{-1})$	$\frac{1}{4\sqrt{2+\sqrt{3}}}(1+z)(1+z^{-1})(1-(2+\sqrt{3})z)(1-(2+\sqrt{3})z^{-1})$
$\frac{1}{8}(1+z)^2(1+z^{-1})$	$\frac{1}{4\sqrt{2+\sqrt{3}}}(1+z^{-1})(1-(2+\sqrt{3})z)(1-(2+\sqrt{3})z^{-1})$
$\frac{1}{8}(1+z)(1+z^{-1})^2$	$\frac{1}{4\sqrt{2+\sqrt{3}}}(1+z)(1-(2+\sqrt{3})z)(1-(2+\sqrt{3})z^{-1})$
$\frac{1}{16}(1+z)^2(1+z^{-1})^2$	$\frac{1}{4\sqrt{2+\sqrt{3}}}(1-(2+\sqrt{3})z)(1-(2+\sqrt{3})z^{-1})$

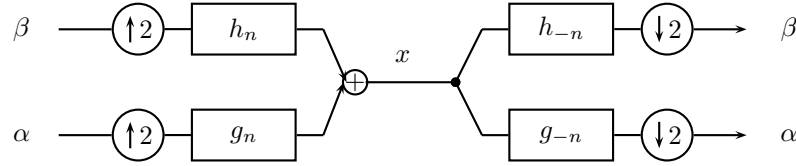
**Table E7.6-2:** Possible choices for the biorthogonal filter pair  $(\tilde{G}(z), G(z))$  in Exercise 7.6.

great importance in practice, since they form the basis for frequency division multiplexing as we show as well. An orthogonal decomposition with many channels leads to orthogonal frequency-division multiplexing (OFDM), the basis for many modulation schemes used in communications, such as 802.11.

We start with

$$X(z) = \begin{bmatrix} G(z) & H(z) \end{bmatrix} \begin{bmatrix} \alpha(z^2) \\ \beta(z^2) \end{bmatrix}. \quad (\text{E7.7-1})$$





**Figure E7.7-1:** Transmultiplexer synthesizes two-channel sequences to a single upsampled sequence, and then splits them again.

The output of the analysis filter bank is

$$\frac{1}{2} \begin{bmatrix} G(z^{-1/2}) & G(-z^{-1/2}) \\ H(z^{-1/2}) & H(-z^{-1/2}) \end{bmatrix} \begin{bmatrix} X(z^{1/2}) \\ X(-z^{1/2}) \end{bmatrix}, \quad (\text{E7.7-2})$$

and we want this to be identical to the inputs  $\alpha(z)$  and  $\beta(z)$ . Substituting (E7.7-1) into this, we see that the input-output relationship is given by the following matrix product (where we formally replaced  $z^{1/2}$  by  $z$ ):

$$\frac{1}{2} \begin{bmatrix} G(z^{-1}) & G(-z^{-1}) \\ H(z^{-1}) & H(-z^{-1}) \end{bmatrix} \begin{bmatrix} G(z) & H(z) \\ G(-z) & H(-z) \end{bmatrix}. \quad (\text{E7.7-3})$$

For this to be identity, we require

$$G(z)G(z^{-1}) + G(-z)G(-z^{-1}) = 2, \quad (\text{E7.7-4a})$$

$$H(z)H(z^{-1}) + H(-z)H(-z^{-1}) = 2, \quad (\text{E7.7-4b})$$

$$G(z)H(z^{-1}) + G(-z)H(-z^{-1}) = 0 \quad (\text{E7.7-4c})$$

$$H(z)G(z^{-1}) + H(-z)G(-z^{-1}) = 0, \quad (\text{E7.7-4d})$$

Of course, the above relations are satisfied if and only if  $G(z)$  and  $H(z)$  are orthogonal filters, since (E7.7-4a) and (E7.7-4b) are orthogonality relations of the filters' impulse responses with respect to their even translates as in (7.13) and (7.14), respectively, while (E7.7-4c) and (E7.7-4d) are the orthogonality relation of  $\{g_n\}$  and  $\{h_{n-2k}\}$  as in (7.22).

Therefore, if we have a two-channel orthogonal filter bank, it does not matter if we cascade analysis followed by synthesis, or synthesis followed by analysis—both will lead to perfect reconstruction systems. The same applies to biorthogonal filter banks, left for Exercise 7.20. An intuitive way to understand the result is to recall that when matrices are square, then the left inverse is also the right inverse.

We now give some geometrical perspective. The output of the synthesis filter bank is the sum of two sequences  $x_V$  and  $x_W$  from  $V = \text{span}(\{g_{n-2k}\}_{k \in \mathbb{Z}})$  and  $W = \text{span}(\{h_{n-2k}\}_{k \in \mathbb{Z}})$ , respectively. Because of the orthogonality of  $g_n$  and  $h_n$ ,

$$V \perp W.$$

## Exercises

### 7.1. FIR Filter Bank

For the system specified by the input-output relation

$$y = GU_2 D_2 \tilde{G}x + HU_2 D_2 \tilde{H}x,$$

- (i) Write the  $z$ -transform of  $y_n$  and  $e^{j\pi n}y_n$  using matrix notation, that is, specify  $\begin{bmatrix} Y(z) & Y(-z) \end{bmatrix}^T$ .

626 Chapter 7. Filter Banks: Building Blocks of Time-Frequency Expansions

- (ii) Assuming that perfect reconstruction can be achieved by this system, show that knowing the analysis filters  $\tilde{G}$ ,  $\tilde{H}$  we can specify the synthesis filters by

$$G(z) = \frac{2z^{-\ell}}{\det(\tilde{\Phi}_m(z))} \tilde{H}(-z) \quad \text{and} \quad H(z) = -\frac{2z^{-\ell}}{\det(\tilde{\Phi}_m(z))} \tilde{G}(-z),$$

where  $\ell$  is any integer and  $\tilde{\Phi}_m(z)$  is called a *modulation matrix*, a matrix of analysis filters,  $\tilde{G}(z)$  and  $\tilde{H}(z)$ , and their modulated versions,  $\tilde{G}(-z)$  and  $\tilde{H}(-z)$ . Specify  $\tilde{\Phi}_m(z)$  and its determinant.

- (iii) If the analysis filters  $\tilde{G}$  and  $\tilde{H}$  are FIR, find a condition that will make the synthesis filters,  $G$  and  $H$ , FIR also. Write the  $z$ -transform of these synthesis filters as a function of the analysis filters.

7.2. *Aliasing Cancellation*

For the system of Exercise 7.1, first find the condition required to cancel all aliasing in the output signal, that is, for the term with  $X(-z)$  to be 0. Then, show that a solution of this condition is given by

$$\tilde{H}(z) = z^{-1} \tilde{G}(-z^{-1}), \quad G(z) = \tilde{G}(z^{-1}), \quad H(z) = z \tilde{G}(-z), \quad (\text{P7.2-1})$$

as proposed by Smith and Barnwell [135]. Finally, using this solution, find the condition for perfect reconstruction as a function of  $\tilde{G}(z)$  in the Fourier domain.

7.3. *Multirate Filtering*

For the system in Figure P7.3-1, find the expressions in both the DTFT and  $z$ -transform domains at each point in the system. Draw the corresponding spectra assuming that  $g$  is an ideal half-band lowpass filter and that  $x$  has the spectrum as in Figure P7.3-2.

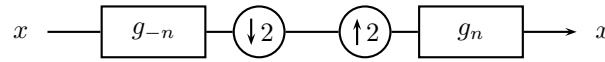


Figure P7.3-1: Multirate system in Exercise 7.3.

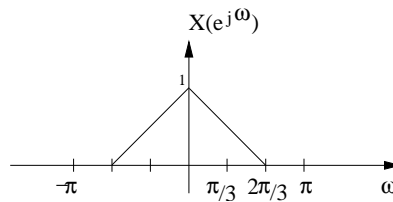


Figure P7.3-2: Spectrum of the input for the multirate system in Exercise 7.3.

7.4. *Rudin-Shapiro Polynomial*

Rudin-Shapiro polynomials are defined by the following recursive equations:

$$\begin{aligned} P_0(z) &= Q_0(z) = 1 \\ P_{n+1}(z) &= P_n(z) + z^{2^n} Q_n(z) \\ Q_{n+1}(z) &= P_n(z) - z^{2^n} Q_n(z) \end{aligned} \quad (\text{P7.4-1})$$

- (i) Derive the Rudin-Shapiro polynomial pair  $(P, Q)$  of degree 15.

- (ii) Prove that for  $n > 0$ ,  $P_n$  and  $Q_n$  lead to a two-channel perfect reconstruction orthogonal filter bank, that is, show the following:

$$\begin{aligned} P_n(z)P_n(z^{-1}) + Q_n(z)Q_n(z^{-1}) &= k_n, \\ P_n(z)P_n(-z^{-1}) + Q_n(z)Q_n(-z^{-1}) &= 0. \end{aligned}$$

Determine the constant  $k_n$ .

- (iii) Prove

$$|P_n(e^{j\omega})|^2 + |Q_n(e^{j\omega})|^2 = 2^{n+1}. \quad (\text{P7.4-2})$$

- (iv) Prove

$$\frac{\mathbb{E}[|P_n(e^{j\omega})|^2]}{\max_{\omega} |P_n(e^{j\omega})|^2} \geq \frac{1}{2},$$

where  $\mathbb{E}[|P_n(e^{j\omega})|^2]$  denotes the average value of  $|P_n(e^{j\omega})|^2$  over one period of length  $2\pi$ .

### 7.5. Sequence Approximation

You are asked to approximate a sequence  $x_n$  with a sequence  $y_n$  having the following form:

$$y_n = \sum_{k \in \mathbb{Z}} \beta_k \phi_{n-2k},$$

where  $\phi_n$  is a length-4 sequence  $\phi = [\dots \ 0 \ \boxed{1} \ 3 \ 3 \ -1 \ 0 \ \dots]$  /  $2\sqrt{5}$ , and  $\beta_n$  is some sequence of coefficients. The DTFT of the original sequence  $x_n$  is  $X(e^{j\omega})$ , and the DTFT of your approximation  $y_n$  is  $Y(e^{j\omega})$ .

- Draw a block-diagram of a system having  $\beta_n$  as input and  $y_n$  as output. Explain what the system does.
- We can compute  $\beta_n = (x_{2n} + 3x_{2n+1} + 3x_{2n+2} - x_{2n+3})/2\sqrt{5}$ . Write an expression for  $Y(e^{j\omega})$  in terms of  $X(e^{j\omega})$ . Draw a block-diagram of a system having  $x_n$  as input and  $y_n$  as output; explain what that system does.
- You wish to find a sequence  $z_n$  which, when added to  $y_n$  exactly recovers  $x_n$ , that is,  $y_n + z_n = x_n$ . For the coefficients  $\beta_n$  in (ii), the sequence  $z_n$  has the same form as  $y_n$ , that is,

$$z_n = \sum_{k \in \mathbb{Z}} \alpha_k \gamma_{n-2k}.$$

Specify the required  $\gamma_n$  sequence, and give an expression for the coefficients  $\alpha_n$  in terms of  $x_n$ .

### 7.6. Zero-Moment Property of Highpass Filters

Verify that a filter with the  $z$ -transform  $H(z) = (1 - z^{-1})^N R(z)$  has its first  $(N - 1)$  moments zero, as in (7.47).

### 7.7. Reproduction of Sinusoids

Consider the proof of Theorem 7.5, about reproduction of polynomials.

- Modify the argument so that complex sinusoids of frequency  $\omega_0$  are reproduced by the lowpass channel.
- Extend the above to real sinusoids of frequency  $\omega_0$ .

### 7.8. Orthogonal Filters Are Maximally Flat

We consider the design of an orthogonal filter with  $N$  zeros at  $z = -1$ . Its deterministic autocorrelation is of the form  $A(z) = (1 + z)^N (1 + z^{-1})^N Q(z)$  as in (7.52) and satisfies (7.53).

- (i) Verify that  $A(e^{j\omega})$  can be written as

$$A(e^{j\omega}) = 2^N (1 + \cos \omega)^N Q(e^{j\omega}).$$

- (ii) Show that  $A(e^{j\omega})$  and its  $(2N - 1)$  derivatives are zero at  $\omega = \pi$ , and show that  $A(e^{j\omega})$  has  $(2N - 1)$  zero derivatives at  $\omega = 0$ .

## 628 Chapter 7. Filter Banks: Building Blocks of Time-Frequency Expansions

- (iii) Show that the previous result leads to  $|G(e^{j\omega})|$  being maximally flat at  $\omega = 0$  and  $\omega = \pi$ , that is,  $|G(e^{j\omega})|$  has  $(N - 1)$  zero derivatives.

## 7.9. Spectral Factorization

For  $C(z) = (1 + z)^3(1 + z^{-1})^3$ , verify that  $D(z) = (3z^2 - 18z + 38 - 18z^{-1} + 3z^{-2})/256$  is such that  $A(z) = C(z)D(z)$  is a valid deterministic autocorrelation. Perform spectral factorization and find the filters of this orthogonal filter bank.

## 7.10. Orthogonal Filter Bank Design

Given is a two-channel FIR filter bank with real coefficients.

- (i) Let  $G(z) = \beta(1 + \alpha z)$  with  $\alpha \in \mathbb{N}_0$  and  $\beta \in \mathbb{R}$ . Give the relation between  $\alpha$  and  $\beta$  such that the filter leads to an orthogonal perfect reconstruction filter bank with real coefficients.
- (ii) Now let  $G(z) = \frac{1}{\sqrt{2}}(1 + 2z)^2$ . Does this filter lead to an orthogonal perfect reconstruction FIR filter bank with real coefficients?
- (iii) Let

$$A(z) = \left(\frac{1 + 2z}{\sqrt{5}}\right)^2 \left(\frac{1 + 2z^{-1}}{\sqrt{5}}\right)^2 R(z).$$

If  $A(z)$  is the  $z$ -transform of a deterministic autocorrelation of a filter, find the shortest polynomial  $R(z)$  such that the associated filter bank leads to perfect reconstruction. Justify why  $R(z)$  cannot be a constant.

- (iv) Using the expression for  $A(z)$ , find a set of four filters (analysis and synthesis) leading to an orthogonal two-channel perfect reconstruction filter bank.

## 7.11. Infinite-Dimensional Bases

Let

$$\begin{aligned} a_n &= \delta_n + 3\delta_{n-1} + 3\delta_{n-2} + \delta_{n-3}, \\ b_n &= \delta_n + 3\delta_{n-1} - 3\delta_{n-2} - \delta_{n-3}. \end{aligned}$$

The set  $\Phi = \{a_{n-2k}, b_{n-2k}\}_{k \in \mathbb{Z}}$  is a basis for  $\ell^2(\mathbb{Z})$ .

- (i) Is  $\Phi$  an orthonormal basis? Why?
- (ii) If yes, demonstrate the Parseval's equality. If no, use Gram-Schmidt orthogonalization to obtain an orthonormal basis for the span of  $\Phi$ .

## 7.12. Complementary Filters

Using Proposition 7.8, prove that  $G(z) = (1 + z^{-1})^N$  always has a complementary  $H(z)$ .

## 7.13. Interpolation Followed by Decimation

Given is the following system: an input  $x$  is upsampled by 2, followed by interpolation with a filter with a  $z$ -transform  $G(z)$  for magnification of the signal. Then, to recover the original signal size, the signal is filtered by  $\tilde{G}(z)$  followed by downsampling by 2, to obtain a reconstruction  $\hat{x}$ .

- (i) What does the product filter  $C(z) = G(z)\tilde{G}(z)$  have to satisfy for  $\hat{x}$  to be a perfect replica of  $x$  (possibly with a shift)?
- (ii) Given  $G(z)$ , what condition does it have to satisfy so that one can find  $\tilde{G}(z)$  achieving perfect reconstruction?
- (iii) For the following two filters,

$$G'(z) = 1 + z^{-1} + z^{-2} + z^{-3}, \quad G''(z) = 1 + z^{-1} + z^{-2} + z^{-3} + z^{-4},$$

give filters  $\tilde{G}'(z)$  and  $\tilde{G}''(z)$  so that perfect reconstruction is achieved (if possible, give shortest such filters; if not, say why).

## 7.14. Structure of Linear-Phase Solutions

Prove the three assertions in Proposition 7.11 on the structure of linear phase solutions.

7.15. *Linear Phase Testing Condition*

Show that, when the filters  $G(z)$  and  $H(z)$  are of the same length and linear phase, the following holds:

$$\Phi_p(z) = z^{-k} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \Phi_p(z^{-1}) \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (\text{P7.15-1})$$

(Hint: Find out the form of the polyphase components of each linear phase filter.)

7.16. *Complex Linear Phase Orthogonal Solutions*

Proposition 7.12 states that there are no real linear-phase orthogonal FIR filter banks.

- (i) Show that if the filter coefficients can be complex valued, then solutions exist.
- (ii) For length-6 filters, find the solution with a maximum numbers of zeros at  $\omega = \pi$ . (Hint: Refactor the  $A(z)$  that leads to the  $D_3$  filter into complex-valued symmetric/antisymmetric filters.)

7.17. *Spectral Factorization Method For Two-Channel Filter Banks*

Consider the factorization of  $P(z)$  in order to obtain orthogonal or biorthogonal filter banks.

- (i) Let

$$A(z) = -\frac{1}{4}z^3 + \frac{1}{2}z + 1 + \frac{1}{2}z^{-1} - \frac{1}{4}z^{-3}$$

Build an orthogonal filter bank based on this  $A(z)$ . If the function is not positive on the unit circle, apply an adequate correction as in Section 7.3.1.

(Hint: This correction, due to Smith and Barnwell, is applied by finding  $\varepsilon = \min A(e^{j\omega})$ . Then take  $A'(z) = (A(z) - \varepsilon)/(1 - \varepsilon)$ .  $A'(z)$  should now satisfy the requirements  $A'(z) + A'(-z) = 2$  and  $A(e^{j\omega}) \geq 0$ .)

- (ii) Alternatively, compute a linear phase factorization of  $A(z)$ . In particular, choose  $G(z) = z + 1 + z^{-1}$ . Give the other filters in this biorthogonal filter bank.
- (iii) Assume now that a particular  $A(z)$  was designed using the Parks-McClellan algorithm (which leads to equiripple pass and stopbands). Show that if  $A(z)$  is not positive on the unit circle, then the correction to make it positive places all stopband zeros on the unit circle.

7.18. *Filter Bank Design Using Lifting*

In a two-channel perfect reconstruction filter bank, FIR solutions are possible if and only if the polyphase matrix of the synthesis bank  $\Phi_p(z)$  is such that its determinant is a pure delay, that is,  $\det \Phi_p(z) = z^{-k}$ .

- (i) Prove that if  $\Phi_p(z)$  meets the above property, then so does  $\Phi'_p(z) = \Phi_p(z)L(z)$  where

$$L(z) = \begin{bmatrix} 1 & R(z) \\ 0 & 1 \end{bmatrix}.$$

- (ii) Give the corresponding diagram for the polyphase implementation of the new synthesis filter bank. What are the corresponding filters  $G'(z)$  and  $H'(z)$ ?
- (iii) Assume  $R(z)$  is an FIR filter of length  $L$  with  $r_0 \neq 0$  and  $r_{L-1} \neq 0$ ,  $L > 1$ . Assume also that the initial filter bank was the Haar orthogonal filter bank. Is the resulting filter bank obtained after lifting orthogonal? In general, if  $\Phi_p(z)$  is any paraunitary matrix, will the resulting filter bank be orthogonal? Justify your answer.

7.19. *Quadrature Mirror Filters*

In this exercise, we explore properties of QMF filter banks.

- (i) Show that the following choice of filters:

analysis	$\tilde{G}(z) = G(z)$	$\tilde{H}(z) = G(-z)$
synthesis	$G(z)$	$H(z) = -G(-z)$

where  $G(z)$  is a linear-phase FIR filter, automatically cancels aliasing in the output.

- (ii) Give the input-output relationship.

---

630 Chapter 7. Filter Banks: Building Blocks of Time-Frequency Expansions

---

- (iii) Explain why QMF FIR filter banks cannot achieve perfect reconstruction.

7.20. *Biorthogonal Transmultiplexer*

Show that a perfect reconstruction analysis-synthesis filter bank is also a perfect reconstruction synthesis-analysis filter bank, by generalizing Solved Exercise 7.7 to biorthogonal filter banks.

7.21. *Frequency-Division Multiplexing with Haar Filters*

Given is a transmultiplexer with Haar filters as in Table 7.8.

- (i) Characterize explicitly the spaces  $V$  and  $W$ , and show that they are orthogonal.
- (ii) Give two example sequences from  $V$  and  $W$ , as well as their sum.
- (iii) Verify explicitly the perfect reconstruction property, either by writing the  $z$ -transform relations or the matrix operators (which is more intuitive and thus preferred).

## Chapter 8

# Local Fourier Bases on Sequences

## Contents

8.1	Introduction . . . . .	632
8.2	$N$ -Channel Filter Banks . . . . .	635
8.3	Complex Exponential-Modulated Local Fourier Bases . . . . .	642
8.4	Cosine-Modulated Local Fourier Bases . . . . .	651
8.5	Computational Aspects . . . . .	661
	Chapter at a Glance . . . . .	663
	Historical Remarks . . . . .	666
	Further Reading . . . . .	666
	Exercises with Solutions . . . . .	666
	Exercises . . . . .	671

Think of a piece of music: notes appear at different instants of time, and then fade away. These are short-time frequency events the human ear identifies easily, but are a challenge for a computer to understand. These notes are well identified frequencies, but they are short lived. Thus, we would like to have access to a *local Fourier transform*, that is, a time-frequency analysis tool that understands the spectrum locally in time. While such a transform is known under many names, such as *windowed Fourier transform*, *Gabor transform* and *short-time Fourier transform*, we will use local Fourier transform exclusively throughout the manuscript. The local energy distribution over frequency, which can be obtained by squaring the magnitude of the local Fourier coefficients, is called the *spectrogram*, and is widely used in speech processing and time-series analysis.

Our purpose in this chapter is to explore what is possible in terms of obtaining such a local version of the Fourier transform of a sequence. While, unfortunately, we will see that, apart from short ones, there exist no good longer local Fourier bases, there exist good *local Fourier frames*, the topic we explore in Chapter 10. Moreover, there exist good *local cosine* bases, where the complex-exponential modulation is

replaced by cosine modulation. These constructions will all be implemented using general,  $N$ -channel filter banks, the first generalization of the basic two-channel filter bank block we just saw in the last chapter.

## 8.1 Introduction

We now look at the simplest example of a local Fourier transform decomposing the spectrum into  $N$  equal parts. As we have learned in the previous chapter, for  $N = 2$ , two-channel filter banks do the trick; for a general  $N$ , it is no surprise that  $N$ -channel filter banks perform that role, and we now show just that.

If we have an infinite-length sequence, we could use the DTFT we discussed in Chapter 2; however, as we mentioned earlier, this representation will erase any time-local information present in the sequence. We could, however, use another tool also discussed in Chapter 2, the DFT. While we have said that the DFT is a natural tool for the analysis of either periodic sequences or infinite-length sequences with a finite number of nonzero samples, circularly extended, we can also use the DFT as a tool to observe the local behavior of an infinite-length sequence by dividing it into pieces of length  $N$ , followed by a length- $N$  DFT.

### Implementing a Length- $N$ DFT Basis Expansion

We now mimic what we have done for the Haar basis in the previous chapter, that is, implement the DFT basis using signal processing machinery. We start with the basis view of the DFT from Section 2.6.1; we assume this finite-dimensional basis is applied to length- $N$  pieces of our input sequence. The final basis then consists of  $\{\varphi_i\}_{i=0}^{N-1}$  from (2.160) and all their shifts by integer multiples of  $N$ , that is,

$$\Phi_{\text{DFT}} = \{\varphi_{i,n-Nk}\}_{i \in \{0,1,\dots,N-1\}, k \in \mathbb{Z}}. \quad (8.1)$$

In other words, as opposed to two template basis sequences generating the entire basis by shifting as in the Haar case, not surprisingly, we now have  $N$  template basis sequences generating the entire basis by shifting. We rename those template basis sequences to (we use the normalized version of the DFT):

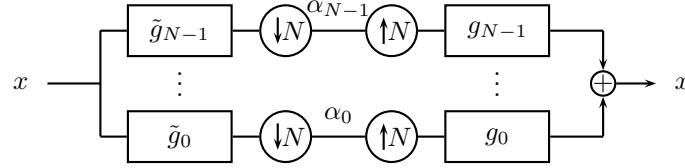
$$g_{i,n} = \varphi_{i,n} = \frac{1}{\sqrt{N}} W_N^{-in}. \quad (8.2)$$

This is again done both for simplicity, as well as because it is the standard way these sequences are denoted.

Then, we rewrite the reconstruction formula (2.159b) as

$$\begin{aligned} x_n &= \sum_{i=0}^{N-1} \sum_{k \in \mathbb{Z}} \underbrace{\langle x_n, \varphi_{i,n-Nk} \rangle_n}_{\alpha_{i,k}} \varphi_{i,n-Nk} \\ &= \sum_{i=0}^{N-1} \sum_{k \in \mathbb{Z}} \alpha_{i,k} \underbrace{\varphi_{i,n-Nk}}_{g_{i,n-Nk}} = \sum_{i=0}^{N-1} \sum_{k \in \mathbb{Z}} \alpha_{i,k} g_{i,n-Nk}, \end{aligned} \quad (8.3a)$$





**Figure 8.1:** An  $N$ -channel analysis/synthesis filter bank.

where we have renamed the basis sequences as explained above, as well as denoted the expansion coefficients as

$$\langle x_n, \varphi_{i,n-Nk} \rangle_n = \langle x_n, g_{i,n-Nk} \rangle_n = \alpha_{i,k}. \quad (8.3b)$$

As for Haar, we recognize each sum in (8.3a) as the output of upsampling by  $N$  followed by filtering ((2.198) for upsampling factor  $N$ ) with the input sequences being  $\alpha_{i,k}$ . Thus, each sum in (8.3a) can be implemented as the input sequence  $\alpha_i$  going through an upsampler by  $N$  followed by filtering by  $g_i$  (right side in Figure 8.1).

By the same token, we can identify the computation of the expansion coefficients  $\alpha_i$  in (8.3b) as (2.195) for downsampling factor  $N$ , that is, filtering by  $g_{i,-n}$  followed by downsampling by  $N$  (left side in Figure 8.1).

We can now merge the above operations to yield an  $N$ -channel filter bank implementing a DFT orthonormal basis expansion as in Figure 8.1. Part on the left, which computes the projection coefficients, is termed an *analysis filter bank*, while the part on the right, which computes the actual projections, is termed a *synthesis filter bank*.

As before, once we have identified all the appropriate multirate components, we can examine the DFT filter bank via matrix operations. For example, in matrix notation, the analysis process (8.3b) can be expressed as

$$\begin{bmatrix} \vdots \\ \boxed{\alpha_{0,0}} \\ \vdots \\ \alpha_{N-1,0} \\ \alpha_{0,1} \\ \vdots \\ \alpha_{N-1,1} \\ \vdots \end{bmatrix} = \frac{1}{\sqrt{N}} \underbrace{\begin{bmatrix} \ddots & & & \\ & F & & \\ & & F & \\ & & & \ddots \end{bmatrix}}_{\Phi^*} \begin{bmatrix} \vdots \\ \boxed{x_0} \\ \vdots \\ x_{N-1} \\ x_N \\ \vdots \\ x_{2N-1} \\ \vdots \end{bmatrix}, \quad (8.4a)$$

with  $F$  as in (2.161a), and the synthesis process (8.3a) as

$$\begin{bmatrix} \vdots \\ \boxed{x_0} \\ \vdots \\ x_{N-1} \\ x_N \\ \vdots \\ x_{2N-1} \\ \vdots \end{bmatrix} = \underbrace{\frac{1}{\sqrt{N}} \begin{bmatrix} \ddots & & & \\ & F^* & & \\ & & F^* & \\ & & & \ddots \end{bmatrix}}_{\Phi} \begin{bmatrix} \vdots \\ \boxed{\alpha_{0,0}} \\ \vdots \\ \alpha_{N-1,0} \\ \alpha_{0,1} \\ \vdots \\ \alpha_{N-1,1} \\ \vdots \end{bmatrix}. \quad (8.4b)$$

Of course,  $\Phi$  is a unitary matrix, since  $F/\sqrt{N}$  is.

**Localization Properties of the Length- $N$  DFT** It is quite clear that the time localization properties of the DFT are superior to those of the DTFT, as now, we have access to the time-local events at the resolution of length  $N$ . However, as a result, the frequency resolution must necessarily worsen; to see this, consider the frequency response of  $g_0$  (the other  $g_k$ s are modulated versions and therefore have the same frequency resolution):

$$G_0(e^{j\omega}) = \sqrt{N} \frac{\text{sinc}(\omega N/2)}{\text{sinc}(\omega/2)}, \quad (8.5)$$

that is, it is the DTFT of a box sequence (see Table 3.6). It has zeros at  $\omega = 2\pi k/N$ ,  $k = 1, 2, \dots, N-1$ , but decays slowly in between.

The orthonormal basis given by the DFT is just one of many basis options implementable by  $N$ -channel filter banks; many others, with template basis sequences with more than  $N$  nonzero samples are possible (similarly to the two-channel case). The DFT is a local Fourier version as the time events can be captured with the resolution of  $N$  samples.

## Chapter Outline

This short introduction leads naturally to the following structure of the chapter: In Section 8.2, we give an overview of  $N$ -channel filter banks. In Section 8.3, we present the local Fourier bases implementable by complex exponential-modulated filter banks. We then come to the crucial, albeit negative result: the Balian-Low theorem, which states the impossibility of good complex exponential-modulated local Fourier bases. We look into their applications: local power spectral density via periodograms, as well as in transmultiplexing. To mitigate the Balian-Low negative result, Section 8.4 considers what happens if we use cosine modulation instead of the complex one to obtain a local frequency analysis. In the block-transform case, we encounter the *discrete cosine transform*, which plays a prominent role in image processing. In the sliding window case, a cosine-modulated filter bank allows

the best of both worlds, namely an orthonormal basis with good time-frequency localization. We also discuss variations on this construction as well as an application to audio compression.

*Notation used in this chapter:* Unlike in the previous chapter, in this one, complex-coefficient filter banks are the norm. Thus, Hermitian transposition is used often, with the caveat that only coefficients should be conjugated and not  $z$ . We will point these out throughout the chapter.  $\square$

## 8.2 $N$ -Channel Filter Banks

We could imagine achieving our goal of splicing the spectrum into  $N$  pieces many ways; we have just seen one, achievable by using the DFT, a representation with reasonable time but poor frequency localization. Another option is using an ideal  $N$ th band filter and its shifts (we have seen it in Tables 2.5 and 3.6, as well as (2.107) with  $\omega_0 = 2\pi/N$ , but repeat it here for completeness):

$$g_{0,n} = \frac{1}{\sqrt{N}} \text{sinc}(\pi n/N) \xleftrightarrow{\text{DTFT}} G_0(e^{j\omega}) = \begin{cases} \sqrt{N}, & |\omega| \leq \pi/N; \\ 0, & \text{otherwise,} \end{cases} \quad (8.6)$$

which clearly has perfect frequency localization but poor time localization as its impulse response is a discrete sinc sequence. We have discussed this trade-off already in Chapter 6, and depict it in Figure 8.2.

The question now is whether there exist constructions in between these two extreme cases? Specifically, are there basis sequences with better frequency localization than the block transform, but with impulse responses that decay faster than the sinc impulse response (for example, a finite impulse response)?

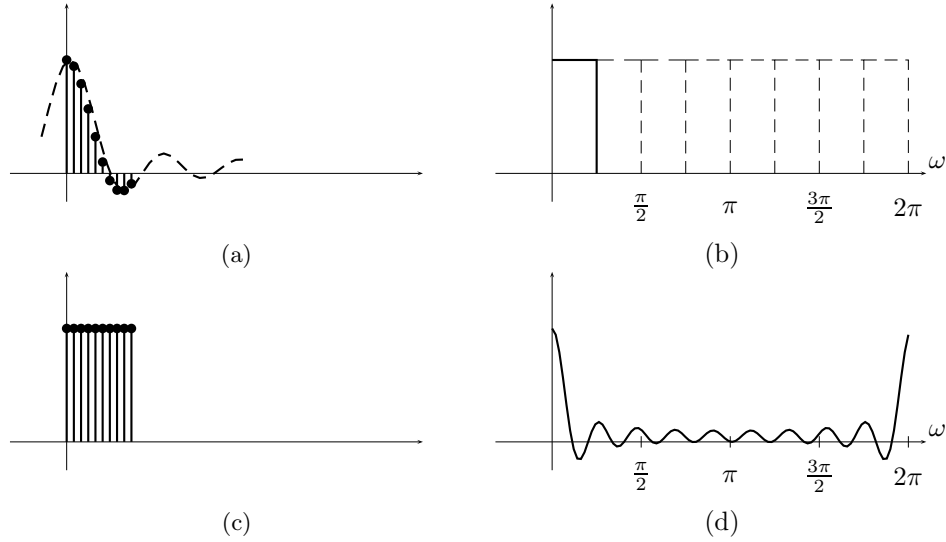
To explore this issue, we introduce general  $N$ -channel filter banks. These are as shown in Figure 8.1, where the input is analyzed by  $N$  filters  $\tilde{g}_i$ ,  $i = 0, 1, \dots, N-1$ , and downsampled by  $N$ . The synthesis is done by upsampling by  $N$ , followed by interpolation with  $g_i$ ,  $i = 0, 1, \dots, N-1$ .

The analysis of  $N$ -channel filter banks can be done in complete analogy to the two-channel case, by using the relevant equations for sampling rate changes by  $N$ . We now state these without proofs, and illustrate them on a particular case of a 3-channel filter bank, especially in polyphase domain.

### 8.2.1 Orthogonal $N$ -Channel Filter Banks

As for two-channel filter banks,  $N$ -channel orthogonal filter banks are of particular interest; the DFT is one example. We now briefly follow the path from the previous chapter and put in one place the relations governing such filter banks. The biorthogonal ones follow similarly, and we just touch upon them during the discussion of the polyphase view.

**Orthogonality of a Single Filter** Since we started with an orthonormal basis, the set  $\{g_{i,n-Nk}\}_{k \in \mathbb{Z}, i \in \{0,1,\dots,N-1\}}$  is an orthonormal set. We have seen in Section 2.7.5



**Figure 8.2:** Time- and frequency-domain behaviors of two orthonormal bases with  $N = 8$  channels. (a)–(b) Sinc basis. (a) Impulse response is a sinc sequence, with poor time localization. (b) Frequency response is a box function, with perfect frequency localization. (c)–(d) DFT basis. (c) Impulse response is a box sequence, with good time localization. (d) Frequency response is a sinc function, with poor frequency localization.

that each such filter is orthogonal and satisfies, analogously to (2.209):

$$\begin{array}{ccc}
 \langle g_{i,n}, g_{i,n-Nk} \rangle = \delta_k & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_N G_i^T G_i U_N = I \\ \sum_{k=0}^{N-1} G_i(W_N^k z) G_i(W_N^{-k} z^{-1}) = N \\ \sum_{k=0}^{N-1} |G_i(e^{j(\omega - (2\pi/N)k)})|^2 = N \end{array} \quad (8.7)
 \end{array}$$

As before, the matrix view expresses the fact that the columns of  $G_i U_N$  form an orthonormal set and the DTFT version is a generalization of the quadrature mirror formula (2.208). For example, take  $g_0$  and  $N = 3$ . The DTFT version is then

$$|G_0(e^{j\omega})|^2 + |G_0(e^{j(\omega - 2\pi/3)})|^2 + |G_0(e^{j(\omega - 4\pi/3)})|^2 = 3;$$

essentially, the magnitude response squared of the filter, added to its modulated versions by  $2\pi/3$  and  $4\pi/3$ , sum up to a constant. This is easily seen in the case of an ideal third-band filter, whose frequency response would be constant  $\sqrt{3}$  (see (8.6)), and thus squared and shifted across the spectrum would satisfy the above.

**Deterministic Autocorrelation of a Single Filter** With  $a_i$  the deterministic autocorrelation of  $g_i$ , the deterministic autocorrelation version of (8.7) is straightfor-

8.2.  $N$ -Channel Filter Banks

637

ward:

$$\begin{array}{ccc}
 \langle g_{i,n}, g_{i,n-Nk} \rangle = a_{i,Nk} = \delta_k & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_N A_i U_N = I \\ \sum_{k=0}^{N-1} A_i(W_N^k z) = N \\ \sum_{k=0}^{N-1} A_i(e^{j(\omega - (2\pi/N)k)}) = N \end{array}
 \end{array} \quad (8.8)$$

**Orthogonal Projection Property a Single Channel** Analogously to two channels, a single channel with orthogonal filters projects onto a coarse subspace  $V_0$  or detail subspaces  $W_i$ ,  $i = 1, 2, \dots, N-1$ , depending on the frequency properties of the filter. Each of the orthogonal projection operators is given as

$$\begin{aligned}
 P_{V_0} &= G_0 U_N D_N G_0^T, \\
 P_{W_i} &= G_i U_N D_N G_i^T, \quad i = 1, 2, \dots, N-1,
 \end{aligned}$$

with the range

$$\begin{aligned}
 V_0 &= \text{span}(\{g_{0,n-Nk}\}_{k \in \mathbb{Z}}), \\
 W_i &= \text{span}(\{g_{i,n-Nk}\}_{k \in \mathbb{Z}}), \quad i = 1, 2, \dots, N-1.
 \end{aligned}$$

**Orthogonality of Filters** As in the previous chapter, once the orthonormality of each single channel is established, what is left is the orthogonality of the channels among themselves. Again, all the expressions are analogous to the two-channel case; we state them here without proof. We assume below that  $i \neq j$ .

$$\begin{array}{ccc}
 \langle g_{i,n}, g_{j,n-Nk} \rangle = 0 & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_N G_j^T G_i U_N = 0 \\ \sum_{k=0}^{N-1} G_i(W_N^k z) G_j(W_N^k z^{-1}) = 0 \\ \sum_{k=0}^{N-1} G_i(e^{j(\omega - (2\pi/N)k)}) G_j(e^{-j(\omega + (2\pi/N)k)}) = 0 \end{array}
 \end{array} \quad (8.9)$$

**Deterministic Crosscorrelation of Filters** Calling  $c_{i,j}$  the deterministic crosscorrelation of  $g_i$  and  $g_j$ :

$$\begin{array}{ccc}
 \langle g_{i,n}, g_{j,n-Nk} \rangle = c_{i,j,Nk} = 0 & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_N C_{i,j} U_N = 0 \\ \sum_{k=0}^{N-1} C_{i,j}(W_N^k z) = 0 \\ \sum_{k=0}^{N-1} C_{i,j}(e^{j(\omega - (2\pi/N)k)}) = 0 \end{array}
 \end{array} \quad (8.10)$$

### 8.2.2 Polyphase View of $N$ -Channel Filter Banks

To cover the polyphase view for general  $N$ , we cover it through an example with  $N = 3$ , and then briefly summarize the discussion for a general  $N$ .

EXAMPLE 8.1 (ORTHOGONAL 3-CHANNEL FILTER BANKS) For two-channel filter banks, a polyphase decomposition is achieved by simply splitting both sequences and filters into their even- and odd-indexed subsequences; for 3-channel filter banks, we split sequences and filters into subsequences modulo 3. While we have seen the expression for a polyphase representation of a sequence and filters for a general  $N$  in (2.222), we write them out for  $N = 3$  to develop some intuition, starting with the input sequence  $x$ :

$$\begin{aligned} x_{0,n} &= x_{3n} \xleftrightarrow{\text{ZT}} X_0(z) = \sum_{n \in \mathbb{Z}} x_{3n} z^{-n}, \\ x_{1,n} &= x_{3n+1} \xleftrightarrow{\text{ZT}} X_1(z) = \sum_{n \in \mathbb{Z}} x_{3n+1} z^{-n}, \\ x_{2,n} &= x_{3n+2} \xleftrightarrow{\text{ZT}} X_2(z) = \sum_{n \in \mathbb{Z}} x_{3n+2} z^{-n}, \\ X(z) &= X_0(z^3) + z^{-1}X_1(z^3) + z^{-2}X_2(z^3). \end{aligned}$$

In the above,  $x_0$  is the subsequence of  $x$  at multiples of 3 downsampled by 3, and similarly for  $x_1$  and  $x_2$ :

$$\begin{aligned} x_0 &= [\dots \ x_{-3} \ \boxed{x_0} \ x_3 \ x_6 \ \dots]^T, \\ x_1 &= [\dots \ x_{-2} \ \boxed{x_1} \ x_4 \ x_7 \ \dots]^T, \\ x_2 &= [\dots \ x_{-1} \ \boxed{x_2} \ x_5 \ x_8 \ \dots]^T. \end{aligned}$$

This is illustrated in Figure 8.3(a): to get  $x_0$  we simply keep every third sample from  $x$ ; to get  $x_1$ , we shift  $x$  by one to the left (advance by one represented by  $z$ ) and then keep every third sample; finally, to get  $x_2$ , we shift  $x$  by two to the left and then keep every third sample. To get the original sequence back, we upsample each subsequence by 3, shift appropriately to the right (delays represented by  $z^{-1}$  and  $z^{-2}$ ), and sum up.

Using (2.222), we define the polyphase decomposition of the synthesis filters:

$$g_{i,0,n} = g_{i,3n} \xleftrightarrow{\text{ZT}} G_{i,0}(z) = \sum_{n \in \mathbb{Z}} g_{i,3n} z^{-n}, \quad (8.11a)$$

$$g_{i,1,n} = g_{i,3n+1} \xleftrightarrow{\text{ZT}} G_{i,1}(z) = \sum_{n \in \mathbb{Z}} g_{i,3n+1} z^{-n}, \quad (8.11b)$$

$$g_{i,2,n} = g_{i,3n+2} \xleftrightarrow{\text{ZT}} G_{i,2}(z) = \sum_{n \in \mathbb{Z}} g_{i,3n+2} z^{-n}, \quad (8.11c)$$

$$G_i(z) = G_{i,0}(z^3) + z^{-1}G_{i,1}(z^3) + z^{-2}G_{i,2}(z^3),$$

where the first subscript denotes the filter, the second the polyphase component, and the last, the discrete time index. In the above, we split each synthesis filter

8.2.  $N$ -Channel Filter Banks

639

into its subsequences modulo 3 as we have done for the input sequence  $x$ :

$$\begin{aligned} g_0 &= [\dots \ g_{-3} \ \boxed{g_0} \ g_3 \ g_6 \ \dots]^T, \\ g_1 &= [\dots \ g_{-2} \ \boxed{g_1} \ g_4 \ g_7 \ \dots]^T, \\ g_2 &= [\dots \ g_{-1} \ \boxed{g_2} \ g_5 \ g_8 \ \dots]^T. \end{aligned}$$

We can now define the *polyphase matrix*  $\Phi_p(z)$ :

$$\Phi_p(z) = \begin{bmatrix} G_{0,0}(z) & G_{1,0}(z) & G_{2,0}(z) \\ G_{0,1}(z) & G_{1,1}(z) & G_{2,1}(z) \\ G_{0,2}(z) & G_{1,2}(z) & G_{2,2}(z) \end{bmatrix}.$$

The matrix above is on the synthesis side; to get it on the analysis side, we define the polyphase decomposition of analysis filters using (2.222) and similarly to what we have done in the two-channel case:

$$\begin{aligned} \tilde{g}_{i,0,n} &= \tilde{g}_{i,3n} = g_{i,-3n} \xleftrightarrow{ZT} \tilde{G}_{i,0}(z) = \sum_{n \in \mathbb{Z}} g_{i,-3n} z^{-n}, \\ \tilde{g}_{i,1,n} &= \tilde{g}_{i,3n-1} = g_{i,-3n+1} \xleftrightarrow{ZT} \tilde{G}_{i,1}(z) = \sum_{n \in \mathbb{Z}} g_{i,-3n+1} z^{-n}, \\ \tilde{g}_{i,2,n} &= \tilde{g}_{i,3n-2} = g_{i,-3n+2} \xleftrightarrow{ZT} \tilde{G}_{i,2}(z) = \sum_{n \in \mathbb{Z}} g_{i,-3n+2} z^{-n}, \\ \tilde{G}(z) &= G_{i,0}(z^{-3}) + zG_{i,1}(z^{-3}) + z^2G_{i,2}(z^{-3}). \end{aligned}$$

The three polyphase components are:

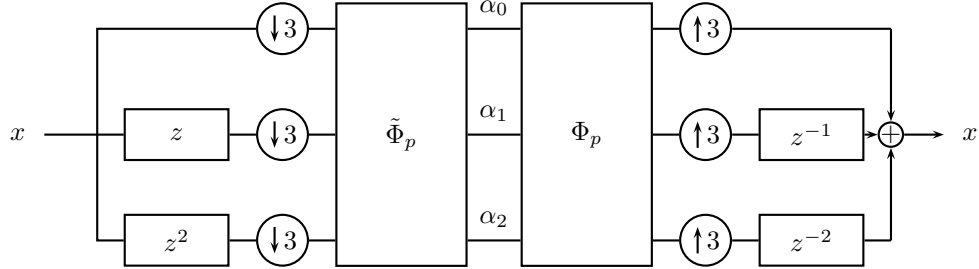
$$\begin{aligned} \tilde{g}_{0,n} &= [\dots \ \tilde{g}_{-3} \ \boxed{\tilde{g}_0} \ \tilde{g}_3 \ \tilde{g}_6 \ \dots]^T = [\dots \ g_3 \ \boxed{g_0} \ g_{-3} \ g_{-6} \ \dots]^T, \\ \tilde{g}_1 &= [\dots \ \tilde{g}_{-4} \ \boxed{\tilde{g}_{-1}} \ \tilde{g}_2 \ \tilde{g}_5 \ \dots]^T = [\dots \ g_4 \ \boxed{g_1} \ g_{-2} \ g_{-5} \ \dots]^T, \\ \tilde{g}_2 &= [\dots \ \tilde{g}_{-5} \ \boxed{\tilde{g}_{-2}} \ \tilde{g}_1 \ \tilde{g}_4 \ \dots]^T = [\dots \ g_5 \ \boxed{g_2} \ g_{-1} \ g_{-4} \ \dots]^T. \end{aligned}$$

Note that  $\tilde{G}_i(z) = G_i(z^{-1})$ , as in the two-channel case. With these definitions, the analysis polyphase matrix is:

$$\tilde{\Phi}_p(z) = \begin{bmatrix} G_{0,0}(z^{-1}) & G_{1,0}(z^{-1}) & G_{2,0}(z^{-1}) \\ G_{0,1}(z^{-1}) & G_{1,1}(z^{-1}) & G_{2,1}(z^{-1}) \\ G_{0,2}(z^{-1}) & G_{1,2}(z^{-1}) & G_{2,2}(z^{-1}) \end{bmatrix} = \Phi_p(z^{-1}).$$

Figure 8.3 shows the polyphase implementation of the system, with the reconstruction of the original sequence using the synthesis polyphase matrix on the right<sup>115</sup> and the computation of projection sequences  $\alpha_i$  on the left; note that

<sup>115</sup>Remember that we typically put the lowpass filter in the lower branch, but in matrices it appears in the first row/column, leading to a slight inconsistency when the filter bank is depicted in the polyphase domain.



**Figure 8.3:** A 3-channel analysis/synthesis filter bank in polyphase domain.

as usual, the analysis matrix (polyphase here) is taken as a transpose (to check it, we could mimic what we did in Section 7.2.4; we skip it here).

The upshot of all this algebra is that we now have a very compact input-output relationship between the input (decomposed into polyphase components) and the result coming out of the synthesis filter bank:

$$X(z) = \begin{bmatrix} 1 & z^{-1} & z^{-2} \end{bmatrix} \Phi_p(z^3) \Phi_p^*(z^{-3}) \begin{bmatrix} X_0(z^3) \\ X_1(z^3) \\ X_2(z^3) \end{bmatrix}.$$

Note that we use the Hermitian transpose here because we will often deal with complex-coefficient filter banks in this chapter. The conjugation is applied only to coefficients and not to  $z$ . The above example went through various polyphase concepts for an orthogonal 3-channel filter bank. We now summarize the same concepts for a general, biorthogonal  $N$ -channel filter bank, and characterize classes of solutions using polyphase machinery.

Using (2.222), in an  $N$ -channel filter bank, the polyphase decomposition of the input sequence, synthesis and analysis filters, respectively, is given by:

$$x_{j,n} = x_{Nn+j} \xleftrightarrow{\text{ZT}} X_j(z) = \sum_{n \in \mathbb{Z}} x_{Nn+j} z^{-n}, \quad (8.12a)$$

$$X(z) = \sum_{j=0}^{N-1} z^{-j} X_j(z^N), \quad (8.12b)$$

$$g_{i,j,n} = g_{i,Nn+j} \xleftrightarrow{\text{ZT}} G_{i,j}(z) = \sum_{n \in \mathbb{Z}} g_{i,Nn+j} z^{-n}, \quad (8.12c)$$

$$G_i(z) = \sum_{j=0}^{N-1} z^{-j} G_{i,j}(z^N), \quad (8.12d)$$



8.2.  $N$ -Channel Filter Banks

641

$$\tilde{g}_{i,j,n} = \tilde{g}_{i,Nn-j} \xleftrightarrow{\text{ZT}} \tilde{G}_{i,j}(z) = \sum_{n \in \mathbb{Z}} \tilde{g}_{i,Nn-j} z^{-n}, \quad (8.12e)$$

$$\tilde{G}_i(z) = \sum_{j=0}^{N-1} z^j \tilde{G}_{i,j}(z^N), \quad (8.12f)$$

leading to the corresponding polyphase matrices:

$$\Phi_p(z) = \begin{bmatrix} G_{0,0}(z) & G_{1,0}(z) & \cdots & G_{N-1,0}(z) \\ G_{0,1}(z) & G_{1,1}(z) & \cdots & G_{N-1,1}(z) \\ \vdots & \vdots & \ddots & \vdots \\ G_{0,N-1}(z) & G_{1,N-1}(z) & \cdots & G_{N-1,N-1}(z) \end{bmatrix}, \quad (8.13a)$$

$$\tilde{\Phi}_p(z) = \begin{bmatrix} \tilde{G}_{0,0}(z) & \tilde{G}_{1,0}(z) & \cdots & \tilde{G}_{N-1,0}(z) \\ \tilde{G}_{0,1}(z) & \tilde{G}_{1,1}(z) & \cdots & \tilde{G}_{N-1,1}(z) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{G}_{0,N-1}(z) & \tilde{G}_{1,N-1}(z) & \cdots & \tilde{G}_{N-1,N-1}(z) \end{bmatrix}. \quad (8.13b)$$

This formulation allows us to characterize classes of solutions. We state these without proof as they follow easily from the equivalent two-channel filter bank results, and can be found in the literature.

**THEOREM 8.1 ( $N$ -CHANNEL FILTER BANKS IN POLYPHASE DOMAIN)** Given is an  $N$ -channel filter bank and the polyphase matrices  $\Phi_p(z)$ ,  $\tilde{\Phi}_p(z)$ . Then:

- (i) The filter bank implements a biorthogonal expansion if and only if

$$\Phi_p(z) \tilde{\Phi}_p^*(z) = I. \quad (8.14a)$$

- (ii) The filter bank implements an orthonormal expansion if and only if

$$\Phi_p(z) \Phi_p^*(z^{-1}) = I, \quad (8.14b)$$

that is,  $\Phi_p(z)$  is paraunitary.

- (iii) The filter bank implements an FIR biorthogonal expansion if and only if  $\Phi_p(z)$  is unimodular (within scaling), that is, if

$$\det(\Phi_p(z)) = \alpha z^{-k}. \quad (8.14c)$$

Note that we use the Hermitian transpose in (8.14b) because we will often deal with complex-coefficient filter banks in this chapter. The conjugation is applied only to coefficients and not to  $z$ .

**Design of  $N$ -Channel Filter Banks** In the next two sections, we will discuss two particular  $N$ -channel filter bank design options, in particular, those that add localization features to the DFT. To design general  $N$ -channel orthogonal filter banks,

we must design  $N \times N$  paraunitary matrices. As in the two-channel case, where such matrices can be obtained by a lattice factorization (see Section 7.3.3),  $N \times N$  paraunitary matrices can be parameterized in terms of elementary matrices ( $2 \times 2$  rotations and delays). Here, we just give an example of a design of a  $3 \times 3$  paraunitary matrix leading to a 3-channel orthogonal filter bank; pointers to literature are given in *Further Reading*.

**EXAMPLE 8.2 (ORTHOGONAL  $N$ -CHANNEL FILTER BANKS)** One way of parameterizing paraunitary matrices is via the following factorization:

$$\Phi_p(z) = U_0 \left[ \prod_{k=1}^{K-1} \text{diag}([z^{-1}, 1, 1]) U_k \right], \quad (8.15a)$$

where

$$U_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_{00} & -\sin \theta_{00} \\ 0 & \sin \theta_{00} & \cos \theta_{00} \end{bmatrix} \begin{bmatrix} \cos \theta_{01} & 0 & -\sin \theta_{01} \\ 0 & 1 & 0 \\ \sin \theta_{01} & 0 & \cos \theta_{01} \end{bmatrix} \begin{bmatrix} \cos \theta_{02} & -\sin \theta_{02} & 0 \\ \sin \theta_{02} & \cos \theta_{02} & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$U_k = \begin{bmatrix} \cos \theta_{k0} & -\sin \theta_{k0} & 0 \\ \sin \theta_{k0} & \cos \theta_{k0} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_{k1} & -\sin \theta_{k1} \\ 0 & \sin \theta_{k1} & \cos \theta_{k1} \end{bmatrix}. \quad (8.15b)$$

The degrees of freedom in design are given by the angles  $\theta_{kj}$ . This freedom in design allows for constructions of orthogonal and linear-phase FIR solutions, not possible in the two-channel case.

### 8.3 Complex Exponential-Modulated Local Fourier Bases

At the start of the previous section, we considered two extreme cases of local Fourier representations implementable by  $N$ -channel filter banks: those based on the DFT (box in time/sinc in frequency, good time/poor frequency localization) and those based on ideal bandpass filters (sinc in time/box in frequency, good frequency/poor time localization). These two particular representations have something in common; as implemented via  $N$ -channel orthogonal filter banks, they are both obtained through a complex-exponential modulation of a single prototype filter.

**Complex-Exponential Modulation** Given a prototype filter  $p = g_0$ , the rest of the filters are obtained via complex-exponential modulation:

$$g_{i,n} = p_n e^{j(2\pi/N)in} = p_n W_N^{-in}, \quad (8.16)$$

$$G_i(z) = P(W_N^i z),$$

$$G_i(e^{j\omega}) = P(e^{j(\omega - (2\pi/N)i)}) = P(W_N^i e^{j\omega}),$$

for  $i = 1, 2, \dots, N-1$ . This is clearly true for the DFT basis from (8.2), (8.5), as well as that constructed from the ideal filters (8.6) (see also Exercise 8.4). A filter bank

## 8.3. Complex Exponential-Modulated Local Fourier Bases

643

implementing such an expansion is often called *complex exponential-modulated filter bank*. While the prototype filter  $p = g_0$  is typically real, the rest of the bandpass filters are complex.

## 8.3.1 Balian-Low Theorem

We are now back to the question whether we can find complex exponential-modulated local Fourier bases with a trade-off of time and frequency localization we have seen for the DFT and sinc bases in Figure 8.2. To that end, we might want to worsen the time localization of the DFT a bit in the hope of improving the frequency one; unfortunately, the following result excludes the possibility of having complex exponential-modulated local Fourier bases with support longer than  $N$ :<sup>116</sup>

**THEOREM 8.2 (DISCRETE BALIAN-LOW THEOREM)** There does not exist a complex exponential-modulated local Fourier basis implementable by an  $N$ -channel FIR filter bank, except for a filter bank with filters of length  $N$ .

*Proof.* To prove the theorem, we analyze the structure of the polyphase matrix of a complex exponential-modulated filter bank with filters as in (8.16). Given the polyphase representation (8.12d) of the prototype filter  $p = g_0$ ,

$$P(z) = P_0(z^N) + z^{-1}P_1(z^N) + \dots + z^{-(N-1)}P_{N-1}(z^N),$$

the modulated versions become

$$G_i(z) = P(W_N^i z) = P_0(z^N) + W_N^{-i} z^{-1} P_1(z^N) + \dots + W_N^{-(N-1)i} z^{-(N-1)} P_{N-1}(z^N)$$

for  $i = 1, 2, \dots, N-1$ . As an example, for  $N = 3$ , the polyphase matrix is

$$\begin{aligned} \Phi_p(z) &= \begin{bmatrix} P_0(z) & P_0(z) & P_0(z) \\ P_1(z) & W_3^{-1}P_1(z) & W_3^{-2}P_1(z) \\ P_2(z) & W_3^{-2}P_2(z) & W_3^{-1}P_2(z) \end{bmatrix} \\ &= \begin{bmatrix} P_0(z) & & \\ & P_1(z) & \\ & & P_2(z) \end{bmatrix} \underbrace{\begin{bmatrix} 1 & 1 & 1 \\ 1 & W_3^{-1} & W_3^{-2} \\ 1 & W_3^{-2} & W_3^{-1} \end{bmatrix}}_{F^*}, \end{aligned} \quad (8.17)$$

that is, a product of a diagonal matrix of prototype filter polyphase components and the conjugated DFT matrix (2.161a). According to Theorem 8.1, this filter bank implements an FIR biorthogonal expansion if and only if  $\Phi_p(z)$  is a monomial. So,

$$\det(\Phi_p(z)) = \prod_{j=0}^{N-1} P_j(z) \underbrace{\det(F^*)}_{1/\sqrt{N}} = \sqrt{N} \prod_{j=0}^{N-1} P_j(z), \quad (8.18)$$

is a monomial if and only if each polyphase component is; in other words, each polyphase component of  $P(z)$  has exactly one nonzero term, or,  $P(z)$  has  $N$  nonzero coefficients (one from each polyphase component).

<sup>116</sup>This result is known as the Balian-Low theorem in the continuous-domain setting, see Section 11.4.1.

While the above theorem is a negative result in general, the proof shows the factorization (8.17) that can be used to derive a fast algorithm, shown in Section 8.5 (the same factorization is used in Solved Exercise 8.2 to derive the relationship between the modulation and polyphase matrices). Rewriting (8.17) for general  $N$ , as well as the analysis polyphase version of it,

$$\Phi_p(z) = \text{diag}([P_0(z), P_1(z), \dots, P_{N-1}(z)]) F^*, \quad (8.19a)$$

$$\tilde{\Phi}_p(z) = \text{diag}([\tilde{G}_{0,0}(z), \tilde{G}_{0,1}(z), \dots, \tilde{G}_{0,N-1}(z)]) F^*, \quad (8.19b)$$

this filter bank implements a basis expansion if and only if

$$\text{diag}([P_0(z), \dots, P_{N-1}(z)]) \text{diag}([\tilde{G}_{0,0}(z), \dots, \tilde{G}_{0,N-1}(z)])^* = z^{-k} N I,$$

a more constrained condition than the general one of  $\Phi_p(z)\tilde{\Phi}_p^*(z) = z^{-k}I$  ( $N$  appears here since we are using the unnormalized version of  $F$ ). We also see exactly what the problem is in trying to have an orthogonal complex exponential-modulated filter bank with filters of length longer than  $N$ : If the filter bank were orthogonal, then  $\tilde{G}_0(z) = G_0(z^{-1}) = P(z^{-1})$ , and the above would reduce to

$$\begin{aligned} \Phi_p(z)\Phi_p^*(z^{-1}) &= \text{diag}([P_0(z), P_1(z), \dots, P_{N-1}(z)]) F^* \\ &\quad F \text{diag}([P_0(z^{-1}), P_1(z^{-1}), \dots, P_{N-1}(z^{-1})])^* \\ &= N \text{diag}([P_0(z)P_0(z^{-1}), P_1(z)P_1(z^{-1}), \dots, P_{N-1}(z)P_{N-1}(z^{-1})]) \\ &= N I, \end{aligned}$$

possible with FIR filters if and only if each polyphase component  $P_j(z)$  of the prototype filter  $P(z)$  were exactly of length 1 (we assumed the prototype to be real). Figure 8.4 depicts a complex exponential-modulated filter bank with  $N = 3$  channels. Solved Exercise 8.2 explores relationships between various matrix representations of a 3-channel complex exponential-modulated filter bank.

### 8.3.2 Application to Power Spectral Density Estimation

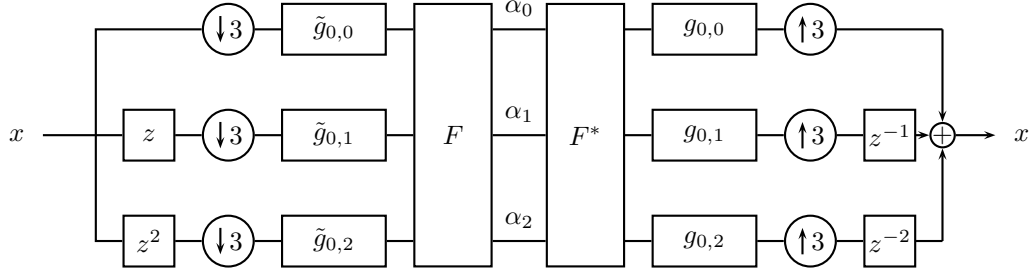
We now discuss the computation of periodograms, a widely used application of complex exponential-modulated filter banks.<sup>117</sup>

Given a discrete stochastic process, there exist various ways to estimate its autocorrelation. In Chapter 2, we have seen that, for a discrete WSS process, there is a direct link between the autocorrelation of a discrete stochastic process and the power spectral density, given by (2.232). However, when the process is changing over time, and we are interested in local behavior, we need a local estimate of the autocorrelation, and therefore, a local power spectral density. We thus need a local Fourier transform, by windowing the sequence appropriately, and squaring the Fourier coefficients to obtain a local power spectral density.

<sup>117</sup>The terms periodogram and spectrogram should not be confused with each other: the former computes the estimate of the power spectral density of a sequence, while the latter shows the dependence of the power spectral density on time.

## 8.3. Complex Exponential-Modulated Local Fourier Bases

645



**Figure 8.4:** A 3-channel analysis/synthesis complex exponential-modulated filter bank, with analysis filters  $\tilde{G}_i(z) = \tilde{G}_0(W_3^i z)$  and synthesis filters  $G_i(z) = G_0(W_3^i z) = P(W_3^i z)$ .  $F$  and  $F^*$  are unnormalized DFT matrices (thus  $3x$  at the output) and are implemented using FFTs.

**Block-Based Power Spectral Density Estimation** A straightforward way to estimate local power spectral density is simply to cut the sequence into adjacent, but nonoverlapping, blocks of size  $M$ ,

$$[\dots x_{-1} \boxed{x_0} x_1 \dots] = [\dots \underbrace{x_{nM} \dots x_{nM+M-1}}_{b_n} \underbrace{x_{(n+1)M} \dots x_{(n+1)M+M-1}}_{b_{n+1}} \dots], \quad (8.20)$$

with  $b_n$  the  $n$ th block of length  $M$ , and then take a length- $M$  DFT of each block,

$$B_n = F b_n, \quad (8.21)$$

with  $F$  from (2.161a). Squaring the magnitudes of the elements of  $B_n$  leads to an approximation of a local power spectral density, known as a *periodogram*.

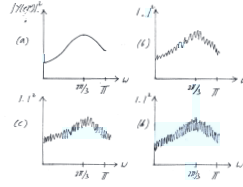
While this method is simple and computationally attractive (using an order  $O(\log M)$  operations per input sample), it has a major drawback we show through a simple example, when the sequence is white noise, or  $x_n$  is i.i.d. with variance  $\sigma_x^2$ . Since  $F$  is a unitary transform (within scaling), or, a rotation in  $M$  dimensions, the entries of  $B_n$  are i.i.d. with variance  $\sigma_x^2$ , independently of  $M$  (see Exercise 8.6). The power spectral density is a constant, but while the resolution increases with  $M$ , the variance does not diminish.

**EXAMPLE 8.3 (BLOCK-BASED POWER SPECTRAL DENSITY ESTIMATION)** Consider a source generated by filtering white Gaussian noise with a causal filter

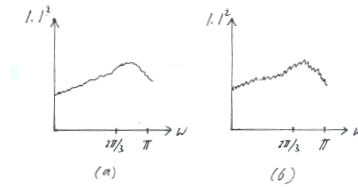
$$H(z) = \frac{1 - \alpha z^{-1}}{(1 - 2\beta \cos \omega_0 z^{-1} + \beta^2 z^{-1})}, \quad \text{ROC} = \{z \mid |z| > \frac{1}{\beta}\}, \quad (8.22)$$

where  $\alpha, \beta$  are real and  $1 < \beta < \infty$ . This filter has poles at  $(1/\beta)e^{\pm j\omega_0}$  and zeroes at  $(1/\alpha)$  and  $\infty$ . The power spectral density of  $y = h * x$  is

$$A_y(e^{j\omega}) = \frac{|1 - \alpha e^{-j\omega}|^2}{|1 - 2\beta \cos \omega_0 e^{-j\omega} + \beta^2 e^{-j2\omega}|^2}, \quad (8.23)$$



**Figure 8.5:** Power spectral density from (8.23). (a) Theoretical, as well as local estimates computed using (8.21) on the blocked version of the source  $y_n$ , with blocks of length (b)  $M = 64$ , (c)  $M = 256$  and (d)  $M = 1024$ , respectively.



**Figure 8.6:** Averaged power spectral density from (8.25). The theoretical power spectral density is the same as in Figure 8.5(a). (a) Average of 16 blocks of length 64. (b) Average of 4 blocks of length 256.

plotted in Figure 8.5(a) for  $\alpha = 1.1$ ,  $\beta = 1.1$  and  $\omega_0 = 2\pi/3$ . Figures 8.5 (b), (c) and (d) show the power spectral density calculated using (8.21) on the blocked version of  $y_n$ , with blocks of length  $M = 64, 256$  and  $1024$ . While the shape of the power spectral density can be guessed, the variance does indeed not diminish.

**Averaged Block-Based Power Spectral Density Estimation** When the sequence is stationary, the obvious fix is to average several power spectra. Calling  $A_{n,k}$  the block-based power spectral density, or, from (8.21)

$$A_{n,k} = |B_{n,k}|^2 = |(Fb_n)_k|^2, \quad (8.24)$$

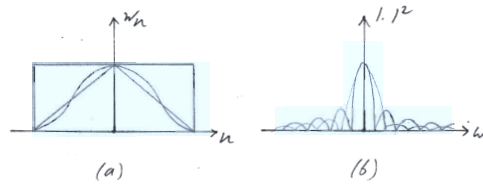
we can define an averaged local power spectrum by summing  $K$  successive ones,

$$A_{n,k}^{(K)} = \frac{1}{K} \sum_{n=0}^{K-1} A_{n,k}, \quad (8.25)$$

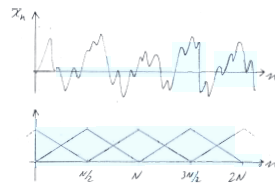
known as an *averaged periodogram*. Exercise 8.6 shows that the variance of  $A_{n,k}^{(K)}$  is about  $1/K$  the variance of  $A_{n,k}$ . Given a length- $L$  ( $L = KM$ ) realization of a stationary process, we can now vary  $M$  or  $K$  to achieve a trade-off between spectral resolution (large  $M$ ) and small variance (large  $K$ ).

## 8.3. Complex Exponential-Modulated Local Fourier Bases

647



**Figure 8.7:** Rectangular, triangular and Hamming windows of length  $M = 31$ , centered at the origin. (a) Time-domain sequences. (b) DTFT magnitude responses (in dB).



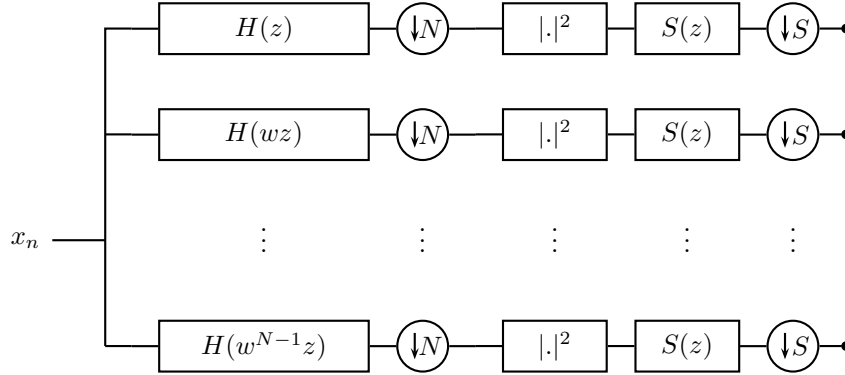
**Figure 8.8:** Windowing with 50% overlap using a triangular window.

#### EXAMPLE 8.4 (AVERAGED BLOCK-BASED POWER SPECTRAL DENSITY ESTIMATION)

Continuing Example 8.3, we now have a realization of length  $L = 1024$ , but would like to reduce the variance of the estimate by averaging. While any factorization of 1024 into  $K$  blocks of length  $1024/K$ , for  $K = 2^i$ ,  $i = 0, 1, \dots, 10$ , is possible, too many blocks lead to a poor frequency resolution ( $K = 1$  was shown in Figure 8.5 (d)). We consider two intermediate cases, 16 blocks of length 64 and 4 blocks of length 256, shown in Figure 8.6(a) and (b). These should be compared to Figure 8.5 (b) and (c), where the same block size was used, but without averaging.

**Estimation Using Windowing and Overlapping Blocks** In practice, both the periodogram and its averaged version are computed using a window, and possibly overlapping blocks.

We first discuss windowing. In the simplest case, the block  $b_n$  in (8.20) corresponds to applying a rectangular window from (2.13a) otherwise, shifted to location  $nM$  (most often the nonunit-norm version, so that the height of the window is 1). To smooth the boundary effects, smoother windows are used, of which many designs are possible. All windows provide a trade-off between the width of the main lobe (the breadth of the DTFT around zero, typically of the order of  $1/M$ ), and the height of the side lobes (the other maxima of the DTFT). Exercise 8.7 considers a few such windows and their respective characteristics. The upshot is that the rectangular window has the narrowest main but the highest side lobe, while others, such as the triangular window, have lower side but broader main lobes. Figure 8.7



**Figure 8.9:** Filter-bank implementation of the periodogram and averaged periodogram using a complex exponential-modulated filter bank with filters (8.16). The sampling factor  $N$  indicates the overlapping between blocks ( $N = M$ , basis;  $N \leq M$ , frame);  $|\cdot|^2$  computes the squared magnitude;  $S(z)$  computes the  $K$ -point averaging filter; and finally, the output is possibly downsampled by  $S$ . (TfBD:  $s$  should be  $S$ ,  $K$  should be  $N$ ,  $N$  should be  $M$ .)

shows three commonly used windows and their DTFT magnitude responses in  $dB$ .

Instead of computing the DFT on adjacent, but nonoverlapping blocks of size  $M$  as in (8.20), we can allow for overlap between adjacent blocks, for example (assume for simplicity  $M$  is even),

$$b_n = [x_{nM/2} \quad x_{nM/2+1} \quad x_{nM/2+2} \quad \cdots \quad x_{nM/2+M-1}], \quad (8.26)$$

a 50% overlap, shown in Figure 8.8, using a triangular window for illustration. In general, the windows move by  $N$  that is smaller or equal to  $M$ .

**Filter-Bank Implementation** The estimation process we just discussed has a natural filter bank implementation. Consider a prototype filter  $\tilde{g}_0 = p$  (we assume a symmetric window so time reversal is not an issue), and construct an  $M$ -channel complex exponential-modulated filter bank as in Figure 8.1 and (8.16). The prototype filter computes the windowing, and the modulation computes the DFT. With the sampling factor  $N = M$ , we get a critically sampled, complex exponential-modulated filter bank. The sampling factor  $N$  can be smaller than  $M$ , in which case the resulting filter bank implements a frame, discussed in Chapter 10 (see Figure 10.14). Squaring the output computes a local approximation to the power spectral density. Averaging the output over  $K$  outputs computes the averaged periodogram, accomplished with a  $K$ -point averaging filter on each of the  $M$  filter bank outputs, or  $S(z) = (1/K) \sum_{m=0}^{K-1} z^{-m}$ . Finally, the output of the averaging filters maybe downsampled by a factor  $S \leq K$ . We have thus constructed a versatile device to compute local power spectral density, summarized in Figure 8.9, and Table 8.1.

The discussion so far focused on nonparametric spectral estimation, that is, we assumed no special structure for the underlying power spectral density. When

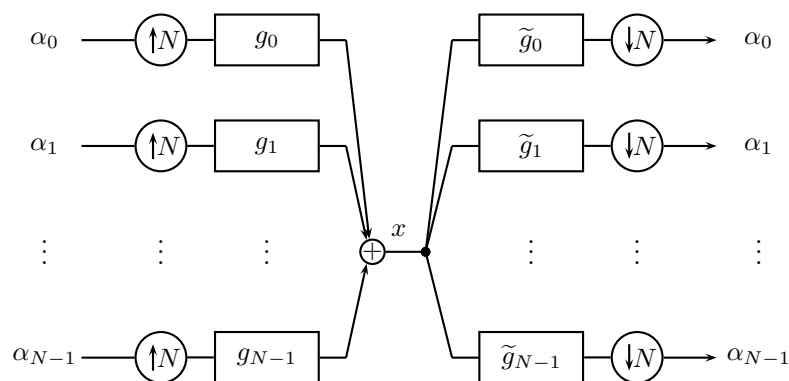


## 8.3. Complex Exponential-Modulated Local Fourier Bases

649

Parameter	Filter-Bank Operation	Computes
$M$	Number of channels	Number of frequency bins (frequency resolution of the analysis)
$\tilde{G}_0(z) = P(z)$	Prototype filter	Windowing
$N$	Downsampling factor	Overlap between adjacent blocks ( $N = M$ , basis; $N < M$ , frame)
$S(z)$	Channel filter	Averaging and variance reduction

**Table 8.1:** Complex exponential-modulated filter-bank implementation of block-based power spectral density estimation.



**Figure 8.10:** A transmultiplexer modulates  $N$  sequences into a single sequence  $x$  of  $N$ -times higher bandwidth, and analyzes it into  $N$  channels.

a deterministic sequence, like a sinusoid, is buried in noise, we have a parametric spectral estimation problem, since we have prior knowledge on the underlying deterministic sequence. While the periodogram can be used here as well (with the effect of windowing now spreading the sinusoid), there exist powerful parametric estimation methods specifically tailored to this problem (see Exercise 8.8).

### 8.3.3 Application to Communications

Transmultiplexers<sup>118</sup> are used extensively in communication systems. They are at the heart of *orthogonal frequency division multiplexing* (OFDM), a modulation scheme popular both in mobile communications as well as in local wireless broadband systems such as IEEE 802.11 (Wi-Fi).

As we have seen in Chapter 7 and Solved Exercise 7.7, a transmultiplexer exchanges the order of analysis/synthesis banks. If the filters used are the complex

<sup>118</sup>Such devices were used to modulate a large number of phone conversations onto large bandwidth transatlantic cables.

exponential-modulated ones as we have seen above, transmultiplexers become computationally efficient. For example, consider  $N$  sequences  $\alpha_i$ ,  $i = 0, 1, \dots, N - 1$ , entering an  $N$ -channel complex exponential-modulated synthesis filter bank, to produce a synthesized sequence  $x$ . Analyzing  $x$  with an  $N$ -channel complex exponential-modulated analysis filter bank should yield again  $N$  sequences  $\alpha_i$ ,  $i = 0, 1, \dots, N - 1$  (Figure 8.10).

Similarly to what we have seen earlier, either length- $N$  filters (from the DFT (8.2)) or sinc filters, (8.6), lead to a basis. Using a good lowpass leads to approximate reconstruction (see Exercise 8.9 for an exploration of the end-to-end behavior).

In typical communication scenarios, a desired signal is sent over a channel with impulse response  $c(t)$  (or equivalently  $c_n$  in a bandlimited and sampled case), and thus, the received signal is the input convolved with the channel. Often, the effect of the channel needs to be canceled, a procedure called *channel equalization*. If the channel is LSI, this equalization can be performed in Fourier domain, assuming  $c_n$  is known (either a priori or measured). This is a first motivation for using a Fourier-like decomposition. Moreover, as the complex sinusoids are eigensignals of LSI systems, such complex sinusoids (or approximations thereof) are good candidates for signaling over a known LSI channel. Namely, an input sinusoid of frequency  $\omega_0$  and a known amplitude  $A$  (or a set of possible amplitudes  $\{A_i\}$ ), will come out of the channel as a sinusoid, scaled by the channel frequency response at  $\omega_0$ , and perturbed by additive channel noise present at that frequency. Digital communication amounts to being able to distinguish a certain number of signaling waveforms per unit of time, given a constraint on the input (such as maximum power).

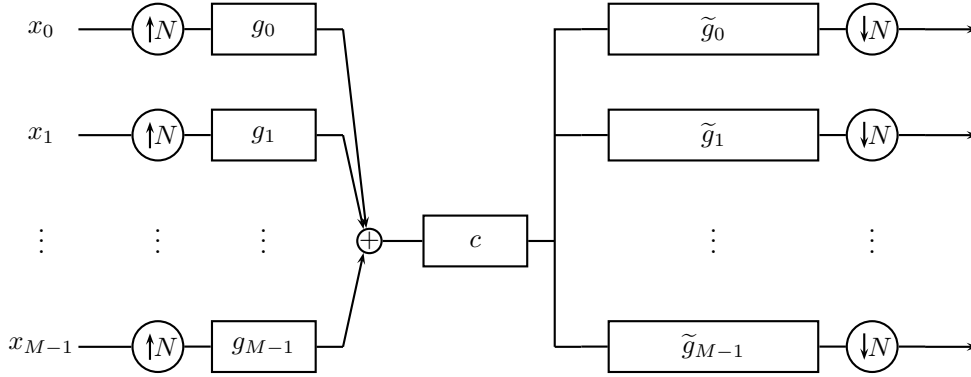
It turns out that an optimal way to communicate over an LSI channel with additive Gaussian noise is precisely to use Fourier-like waveforms. While an ideal system would require a very large number of perfect bandpass channels, practical systems use a few hundred channels (for example, 256 or 512). Moreover, instead of perfect bandpass filters (which require sinc filters), approximate bandpass filters based on finite windows are used. This time localization also allows to adapt to a changing channel, for example, in mobile communications.

The system is summarized in Figure 8.11, for  $M$  channels upsampled by  $N$ .<sup>119</sup> Such a device, allowing to put  $M$  sequences  $\{x_i\}_{i=0,\dots,M-1}$ , onto a single channel of  $N$ -times larger bandwidth, has been historically known as a transmultiplexer.

When the prototype filter is a rectangular window of length  $M$ , in the absence of channel effects, the synthesis/analysis complex exponential-modulated filter bank is perfect reconstruction. When a channel is present, and the prototype filter is a perfect lowpass filter of bandwidth  $[-\pi/M, \pi/M]$ , each bandpass channel is affected by the channel independently of the others, and can be individually equalized.

For filters with finite impulse response, one can either use a narrower-band prototype (so neighboring channels do not interact), or, use fewer than the critical number of channels, which in both cases means some redundancy is left in the synthesized sequence that enters the channel.

<sup>119</sup>Again, if  $M > N$ , such a filter bank implements a frame, discussed in Chapter 10.



**Figure 8.11:** Communication over a channel using a complex exponential-modulated transmultiplexer. When  $M = N$ , it is critically sampled, while for  $M > N$ , the sequence entering the channel is redundant.

## 8.4 Cosine-Modulated Local Fourier Bases

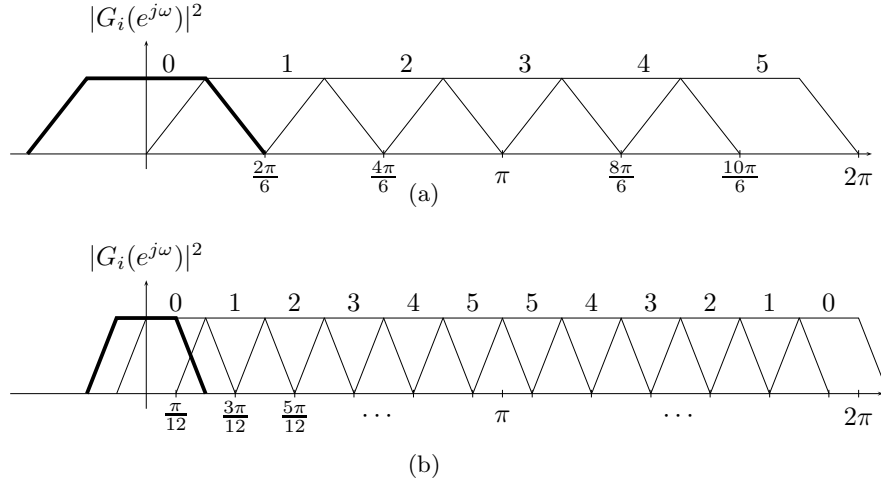
A possible escape from the restriction imposed by the Balian-Low theorem is to replace complex-exponential modulation (multiplication by  $W_N^i = e^{-j2\pi i/N}$ ) with an appropriate cosine modulation. This has an added advantage that all filters are real if the prototype is real.

**Cosine Modulation** Given a prototype filter  $p$ , all of the filters are obtained via cosine modulation:

$$\begin{aligned}
 g_{i,n} &= p_n \cos\left(\frac{2\pi}{2N}\left(i + \frac{1}{2}\right)n + \theta_i\right) \\
 &= p_n \frac{1}{2} \left[ e^{j\theta_i} W_{2N}^{-(i+1/2)n} + e^{-j\theta_i} W_{2N}^{(i+1/2)n} \right], \\
 G_i(z) &= \frac{1}{2} \left[ e^{j\theta_i} P(W_{2N}^{(i+1/2)} z) + e^{-j\theta_i} P(W_{2N}^{-(i+1/2)} z) \right], \\
 G_i(e^{j\omega}) &= \frac{1}{2} \left[ e^{j\theta_i} P(e^{j(\omega - (2\pi/2N)(i+1/2))}) + e^{-j\theta_i} P(e^{j(\omega + (2\pi/2N)(i+1/2))}) \right],
 \end{aligned} \tag{8.27}$$

for  $i = 0, 1, \dots, N-1$ , and  $\theta_i$  is a phase factor that gives us flexibility in designing the representation; you may assume it to be 0 for now. Compare the above with (8.16) for the complex-exponential modulation; the difference is that given a real prototype filter, all the other filters are real. Moreover, the effective bandwidth, while  $2\pi/N$  in the case of complex-exponential modulation, is  $\pi/N$  here. The difference occurs because, the cosine-modulated filters being real, have two side lobes, which reduces the bandwidth per side lobe by two. The modulation frequencies follow from an even coverage of the interval  $[0, \pi]$  with side lobes of width  $\pi/N$ . This is illustrated in Figure 8.12 for  $N = 6$  for both complex as well as cosine modulation.

Will such a modulation lead to an orthonormal basis? One possibility is to



**Figure 8.12:** Complex exponential-modulated versus cosine-modulated filter bank with  $N = 6$ . (a) In the complex case, the bandwidth of the prototype is  $2\pi/6$ , and the center frequencies are  $2\pi i/6$ ,  $i = 0, 1, \dots, 5$ . (b) In the cosine case, the bandwidth of the prototype is  $2\pi/12$ , and the center frequencies are  $(2i + 1)\pi/12$ ,  $i = 0, 1, \dots, 5$ . Unlike in (a), the first filter does not correspond to the prototype, but is modulated to  $\pi/12$ .

choose an ideal lowpass filter as prototype, with support  $[-\pi/2N, \pi/2N]$ . However, as we know, this leads to a sinc-like basis with infinite and slowly-decaying impulse responses. Another solution is a block transform, such as the *discrete cosine transform (DCT)* discussed in Solved Exercise 8.3, too short for an interesting analysis. Fortunately, other solutions exist, with FIR filters of length longer than  $N$ , which we introduce next.

### 8.4.1 Lapped Orthogonal Transforms

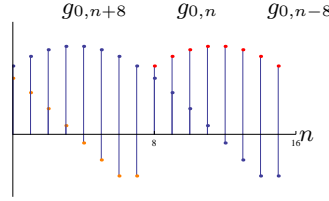
The earliest example of such cosine-modulated bases was developed for filters of length  $2N$ , implying that the nonzero support of the basis sequences overlaps by  $N/2$  on each side of a block of length  $N$  (see Figure 8.13), earning them the name *lapped orthogonal transforms (LOT)*.

#### LOTs with a Rectangular Prototype Window

Consider  $N$  filters  $g_0, g_1, \dots, g_{N-1}$  of length  $2N$  given by (8.27). We start with a rectangular prototype window filter:

$$p_n = \frac{1}{\sqrt{N}}, \quad n = 0, 1, \dots, 2N - 1, \quad (8.28)$$

where, by choosing  $p_n = 1/\sqrt{N}$ , we ensured that  $\|g_i\| = 1$ , for all  $i$ .



**Figure 8.13:** LOT for  $N = 8$ . The filters are of length  $2N = 16$ , and thus, they overlap with their nearest neighbors by  $N/2 = 4$  (only  $g_0$  is shown). Tails are orthogonal since the left half (red) is symmetric and the right half (orange) is antisymmetric.

As we always do, we first find conditions so that the set  $\{g_{i,n-Nk}\}$ ,  $k \in \mathbb{Z}$ ,  $i \in \{0, 1, \dots, N-1\}$ , is an orthonormal set. To do that, we prove (8.7) and (8.9).

**Orthogonality of a Single Filter** To prove that a single filter  $g_i$  is orthogonal to its shifts by  $N$ , it is enough to prove this for just two neighboring shifts (as the length of the filters is  $2N$ , see Figure 8.13). An easy way to force this orthogonality would be if the left half (tail) of the filter support (from  $0, 1, \dots, N-1$ ) were symmetric around its midpoint  $(N-1)/2$ , while the right half (tail) of the filter support (from  $N, N+1, \dots, 2N-1$ ) were antisymmetric around its midpoint  $(3N-1)/2$ . Then, the inner product  $\langle g_{i,n}, g_{i,n-N} \rangle$  would amount to the inner product of the right tail of  $g_{i,n}$  with the left tail of  $g_{i,n-N}$ , and would automatically be zero as a product of a symmetric sequence with an antisymmetric sequence. The question is whether we can force such conditions on all the filters. Fortunately, we have a degree of freedom per filter given by  $\theta_i$ , which we choose to be

$$\theta_i = -\frac{2\pi}{2N} \left(i + \frac{1}{2}\right) \frac{N-1}{2}. \quad (8.29)$$

After substituting it into (8.27), we get

$$g_{i,n} = \frac{1}{\sqrt{N}} \cos \left( \frac{2\pi}{2N} \left(i + \frac{1}{2}\right) \left(n - \frac{N-1}{2}\right) \right). \quad (8.30)$$

We now check the symmetries of the tails; for the left tail,

$$g_{i,N-n-1} = \frac{1}{\sqrt{N}} \cos \left( \frac{2\pi}{2N} \left(i + \frac{1}{2}\right) \left(-n + \frac{N-1}{2}\right) \right) \stackrel{(a)}{=} g_{i,n}, \quad (8.31a)$$

for  $n = 0, 1, \dots, N/2 - 1$ , that is, it is indeed symmetric. In the above, (a) follows from the symmetry of the cosine function. Similarly, for the right tail,

$$\begin{aligned}
 g_{i,2N-n-1} &= \frac{1}{\sqrt{N}} \cos\left(\frac{2\pi}{2N} \left(i + \frac{1}{2}\right) \left(-n + \frac{3N-1}{2}\right)\right) \\
 &\stackrel{(a)}{=} \frac{1}{\sqrt{N}} \cos\left(\frac{2\pi}{2N} \left(i + \frac{1}{2}\right) \left(n - \frac{3N-1}{2}\right)\right) \\
 &= \frac{1}{\sqrt{N}} \cos\left(\frac{2\pi}{2N} \left(i + \frac{1}{2}\right) \left(n + \frac{N+1}{2} - 2N\right)\right) \\
 &= \frac{1}{\sqrt{N}} \cos\left(\frac{2\pi}{2N} \left(i + \frac{1}{2}\right) \left(n + \frac{N+1}{2}\right) + \pi\right) \\
 &\stackrel{(b)}{=} -g_{i,N+n},
 \end{aligned} \tag{8.31b}$$

for  $n = 0, 1, \dots, N/2 - 1$ , that is, it is indeed antisymmetric. In the above, (a) follows from the symmetry of the cosine function and (b) from  $\cos(\theta + \pi) = -\cos(\theta)$ . An LOT example with  $N = 8$  is given in Figure 8.14.

**Orthogonality of Filters** We now turn our attention to showing that all the filters are orthogonal to each other (and their shifts). As we have done in (8.27), we use (2.275), to express  $g_i$  from (8.30)

$$g_{i,n} = \frac{1}{2\sqrt{N}} \left( W_{2N}^{(i+1/2)(n-(N-1)/2)} + W_{2N}^{-(i+1/2)(n-(N-1)/2)} \right). \tag{8.32}$$

The inner product between two different filters is then:

$$\begin{aligned}
 \langle g_i, g_k \rangle &= \frac{1}{4N} \sum_{n=0}^{2N-1} \left( W_{2N}^{(i+1/2)(n-(N-1)/2)} + W_{2N}^{-(i+1/2)(n-(N-1)/2)} \right) \\
 &\quad \left( W_{2N}^{(k+1/2)(n-(N-1)/2)} + W_{2N}^{-(k+1/2)(n-(N-1)/2)} \right) \\
 &= \frac{1}{4N} \sum_{n=0}^{2N-1} \left( W_{2N}^{(i+k+1)(n-(N-1)/2)} + W_{2N}^{(i-k)(n-(N-1)/2)} + \right. \\
 &\quad \left. W_{2N}^{-(i-k)(n-(N-1)/2)} + W_{2N}^{-(i+k+1)(n-(N-1)/2)} \right).
 \end{aligned}$$

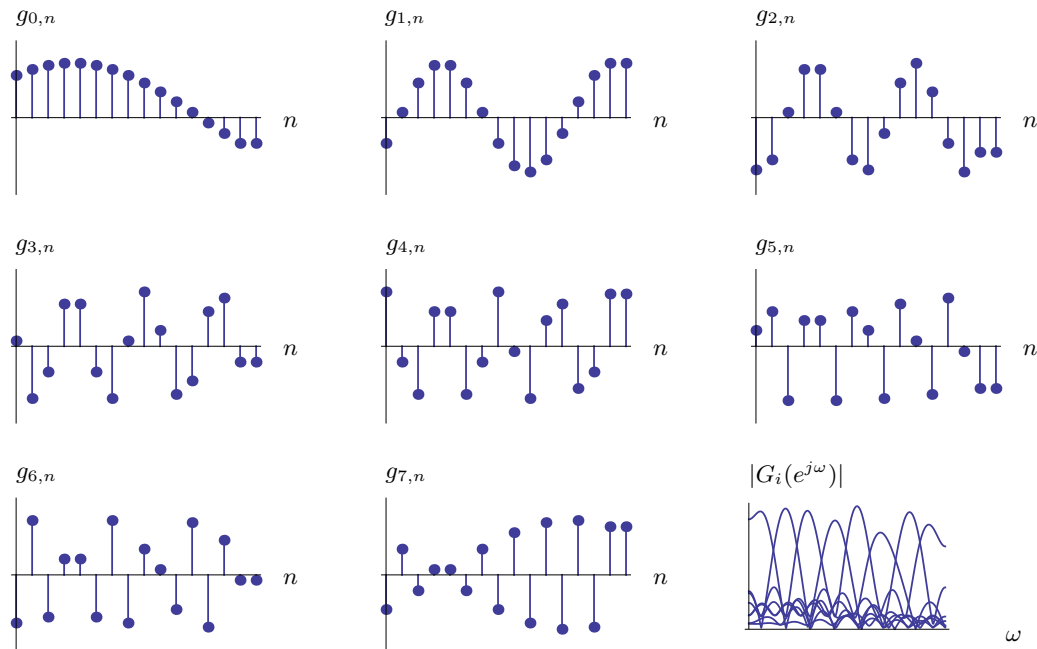
To show that the above inner product is zero, we show that each of the four sums are zero. We show it for the first sum; the other three follow the same way.

$$\frac{1}{4N} \sum_{n=0}^{2N-1} W_{2N}^{(i+k+1)(n-(N-1)/2)} = \frac{1}{4N} W_{2N}^{-(N-1)/2} \sum_{n=0}^{2N-1} (W_{2N}^{(i+k+1)})^n = 0, \tag{8.33}$$

because of the orthogonality of the roots of unity (2.277c).

## 8.4. Cosine-Modulated Local Fourier Bases

655



**Figure 8.14:** LOT for  $N = 8$  with a rectangular prototype window. The eight basis sequences (note the symmetric and antisymmetric tails) and their magnitude responses, showing the uniform split of the spectrum.

**Matrix View** As usual, we find the matrix view of an expansion to be illuminating. We can write the output of an LOT synthesis bank similarly to (7.7),

$$\Phi = \begin{bmatrix} \ddots & & & & & \\ & G_0 & & & & \\ & G_1 & G_0 & & & \\ & & G_1 & G_0 & & \\ & & & G_1 & G_0 & \\ & & & & \ddots & \end{bmatrix}, \quad (8.34)$$

where the columns of  $G_0$  and  $G_1$  are the left and right tails respectively,

$$\begin{bmatrix} G_0 \\ G_1 \end{bmatrix} = \begin{bmatrix} g_{0,0} & g_{1,0} & \cdots & g_{N-1,0} \\ g_{0,1} & g_{1,1} & \cdots & g_{N-1,1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{0,N-1} & g_{1,N-1} & \cdots & g_{N-1,N-1} \\ g_{0,N} & g_{1,N} & \cdots & g_{N-1,N} \\ g_{0,N+1} & g_{1,N+1} & \cdots & g_{N-1,N+1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{0,2N-1} & g_{1,2N-1} & \cdots & g_{N-1,2N-1} \end{bmatrix}. \quad (8.35)$$

Since the expansion is orthonormal,  $\Phi\Phi^T = I$ , but also  $\Phi^T\Phi = I$ , or,

$$G_0 G_0^T + G_1 G_1^T = I, \quad (8.36a)$$

$$G_1 G_0^T = G_0 G_1^T = 0, \quad (8.36b)$$

$$G_0^T G_0 + G_1^T G_1 = I, \quad (8.36c)$$

$$G_0^T G_1 = G_1^T G_0 = 0. \quad (8.36d)$$

Following the symmetry/antisymmetry of the tails, the matrices  $G_0$  and  $G_1$  have repeated rows. For example, for  $N = 4$ ,

$$G_0 = \begin{bmatrix} g_{0,0} & g_{1,0} & g_{2,0} & g_{3,0} \\ g_{0,1} & g_{1,1} & g_{2,1} & g_{3,1} \\ g_{0,1} & g_{1,1} & g_{2,1} & g_{3,1} \\ g_{0,0} & g_{1,0} & g_{2,0} & g_{3,0} \end{bmatrix} \quad \text{and} \quad G_1 = \begin{bmatrix} g_{0,4} & g_{1,4} & g_{2,4} & g_{3,4} \\ g_{0,5} & g_{1,5} & g_{2,5} & g_{3,5} \\ -g_{0,5} & -g_{1,5} & -g_{2,5} & -g_{3,5} \\ -g_{0,4} & -g_{1,4} & -g_{2,4} & -g_{3,4} \end{bmatrix}.$$

Denoting by  $\hat{G}_0$  and  $\hat{G}_1$  the upper halves of  $G_0$  and  $G_1$ , respectively, we can express  $G_0$  and  $G_1$  as

$$G_0 = \begin{bmatrix} I_{N/2} \\ J_{N/2} \end{bmatrix} \hat{G}_0 \quad \text{and} \quad G_1 = \begin{bmatrix} I_{N/2} \\ -J_{N/2} \end{bmatrix} \hat{G}_1, \quad (8.37)$$

where  $I_{N/2}$  is an  $N/2 \times N/2$  identity matrix and  $J_{N/2}$  is an  $N/2 \times N/2$  antidiagonal matrix (defined in Section 1.B.2). Note that  $J_N^2 = I_N$ , and that premultiplying by  $J_N$  reverses the row order (postmultiplying reverses the column order).

From the above, both  $G_0$  and  $G_1$  have rank  $N/2$ . We can easily check that the rows of  $\hat{G}_0$  and  $\hat{G}_1$  form an orthogonal set, with norm  $1/\sqrt{2}$ . Using all of the above, we finally unearth the special structure of the LOTs:

$$\begin{aligned} G_0 G_0^T &= \begin{bmatrix} I_{N/2} \\ J_{N/2} \end{bmatrix} \hat{G}_0 \hat{G}_0^T \begin{bmatrix} I_{N/2} & J_{N/2} \end{bmatrix} = \begin{bmatrix} I_{N/2} \\ J_{N/2} \end{bmatrix} \frac{1}{2} I_{N/2} \begin{bmatrix} I_{N/2} & J_{N/2} \end{bmatrix} \\ &= \frac{1}{2} \begin{bmatrix} I_{N/2} & J_{N/2} \\ J_{N/2} & I_{N/2} \end{bmatrix} = \frac{1}{2} (I_N + J_N), \end{aligned} \quad (8.38a)$$

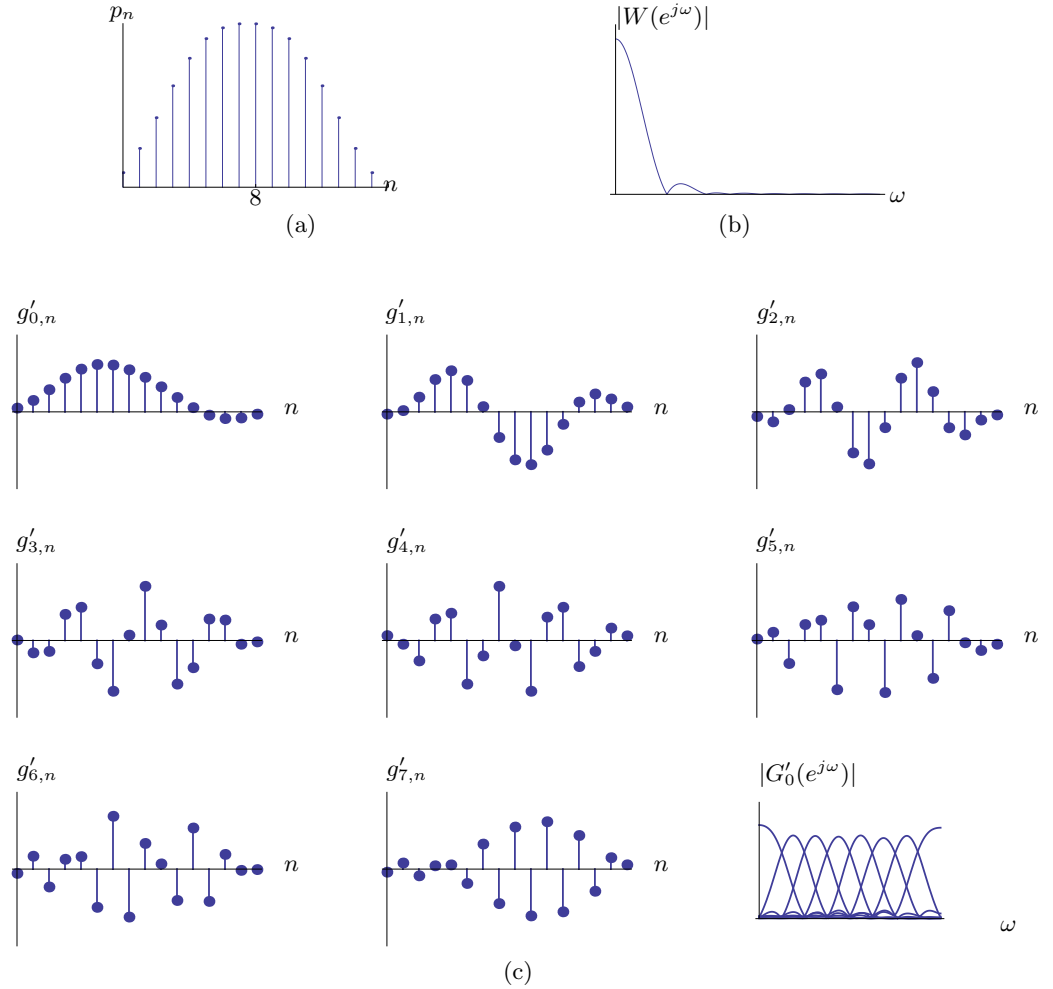
$$G_1 G_1^T = \frac{1}{2} \begin{bmatrix} I_{N/2} & -J_{N/2} \\ -J_{N/2} & I_{N/2} \end{bmatrix} = \frac{1}{2} (I_N - J_N), \quad (8.38b)$$

$$G_0^T G_1 = G_1^T G_0 = 0. \quad (8.38c)$$



## 8.4. Cosine-Modulated Local Fourier Bases

657



**Figure 8.15:** LOT for  $N = 8$  with a smooth, power-complementary prototype window from Table 8.2. Its (a) impulse response and (b) magnitude response. (c) The eight windowed basis sequences and their magnitude responses. Note the improved frequency resolution compared to Figure 8.14(b).

### LOTs with a Nonrectangular Prototype Window

At this point, we have  $N$  filters of length  $2N$ , but their impulse responses are simply rectangularly-windowed cosine sequences. Such a rectangular prototype window is discontinuous at the boundary, and thus not desirable; instead we aim for smooth tapering at the boundary. We now investigate whether we can window our previous solution with a smooth prototype window and still retain orthogonality.

$p_0$	$p_1$	$p_2$	$p_3$	$p_4$	$p_5$	$p_6$	$p_7$
0.0887655	2366415	0.4238081	0.6181291	0.7860766	0.9057520	0.9715970	0.9960525

**Table 8.2:** Power-complementary prototype window used in Figure 8.15. The prototype window is symmetric, so only half of the coefficients are shown.

For this, we choose a power-complementary,<sup>120</sup> real and symmetric prototype window sequence  $p$ , such that:

$$p_{2N-n-1} = p_n, \quad (8.39a)$$

$$|p_n|^2 + |p_{N-n-1}|^2 = 2, \quad (8.39b)$$

for  $n = 0, 1, \dots, N-1$ . Let

$$\begin{aligned} P_0 &= \text{diag}([p_0, p_1, \dots, p_{N-1}]), \\ P_1 &= \text{diag}([p_N, p_{N+1}, \dots, p_{2N-1}]). \end{aligned}$$

Then, (8.39) can be rewritten as

$$P_1 = J_N P_0 J_N, \quad (8.40a)$$

$$P_0^2 + P_1^2 = 2I. \quad (8.40b)$$

The counterpart to (8.35) are now the windowed impulse responses

$$\begin{bmatrix} G'_0 \\ G'_1 \end{bmatrix} = \begin{bmatrix} P_0 G_0 \\ P_1 G_1 \end{bmatrix} = \begin{bmatrix} P_0 & \\ & J_N P_0 J_N \end{bmatrix} \begin{bmatrix} G_0 \\ G_1 \end{bmatrix}. \quad (8.41)$$

Note that

$$P_0 J_N P_0 = P_1 J_N P_1. \quad (8.42)$$

These windowed impulse responses have to satisfy (8.36), or, (8.38) (substituting  $G'_i$  for  $G_i$ ). For example, we check the orthogonality of the tails (8.38c):

$$G_1'^T G_0' \stackrel{(a)}{=} G_1^T (J_N P_0 J_N) P_0 G_0 \stackrel{(b)}{=} \hat{G}_1^T \begin{bmatrix} I_{N/2} & -J_{N/2} \end{bmatrix} J_N P_0 J_N P_0 \begin{bmatrix} I_{N/2} \\ J_{N/2} \end{bmatrix} \hat{G}_0,$$

where (a) follows from (8.41), and (b) from (8.37). As the product  $J_N P_0 J_N P_0$  is diagonal and symmetric (the  $k$ th entry is  $p_k p_{N-k}$ ), we get

$$\begin{bmatrix} I_{N/2} & -J_{N/2} \end{bmatrix} J_N P_0 J_N P_0 \begin{bmatrix} I_{N/2} \\ J_{N/2} \end{bmatrix} = 0.$$

<sup>120</sup>The term *power complementary* is typically used to denote a filter whose magnitude response squared added to the frequency-reversed version of the magnitude response squared, sums to a constant, as in (2.208). We use the term more broadly here to denote a sequence whose magnitude squared added to the time-reversed version of the magnitude squared, sums to a constant.

## 8.4. Cosine-Modulated Local Fourier Bases

659

To complete the orthogonality proof, we need to verify (8.36a) (with appropriate substitutions as above),

$$\begin{aligned}
 G'_0(G'_0)^T + G'_1(G'_1)^T &\stackrel{(a)}{=} P_0 G_0 G_0^T P_0 + P_1 G_1 G_1^T P_1 \\
 &\stackrel{(b)}{=} \frac{1}{2} P_0 (I_N + J_N) P_0 + \frac{1}{2} P_1 (I_N - J_N) P_1 \\
 &= \underbrace{\frac{1}{2} (P_0^2 + P_1^2)}_I + \underbrace{\frac{1}{2} (P_0 J_N P_0 - P_1 J_N P_1)}_0 \stackrel{(c)}{=} I,
 \end{aligned}$$

where (a) follows from (8.41); (b) from (8.38) and (c) from (8.40b) and (8.42). An example of a windowed LOT is shown in Figure 8.15 for  $N = 8$ . The prototype window is symmetric of length 16, with coefficients as in Table 8.2.

**Shift-Varying LOT Filter Banks** We end this section with a discussion of a variation on the theme of prototype windows, both for its importance in practice<sup>121</sup> and because it shows the same basic principles at work. Assume one wants to process a sequence with an  $N$ -channel filter bank and then switch to a  $2N$ -channel filter bank. In addition, one would like a smooth rather than an abrupt transition. Interestingly, to achieve this, it is enough for the two adjacent prototype windows to have overlapping tails that are power complementary (see Figure 8.16). Calling  $p^{(L)}$  and  $p^{(R)}$  the two prototype windows involved, then

$$|p_n^{(L)}|^2 + |p_n^{(R)}|^2 = 2$$

leads again to orthogonality of the overlapping tails of the two filter banks.

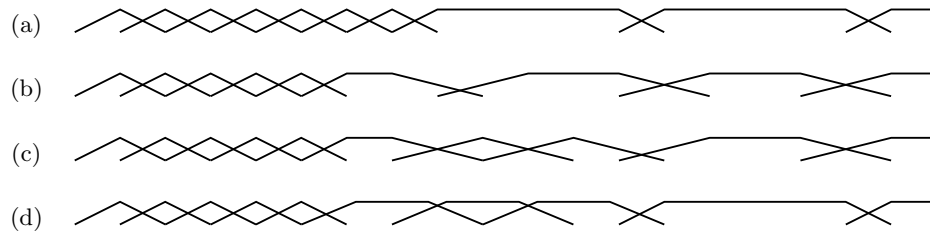
## 8.4.2 Application to Audio Compression

In Section 8.3.2, we have made numerous references to redundancy, which we will discuss in Chapter 10. In compression, the opposite is required: we want to remove the redundancy from the sequence as much as possible, and thus, typically, bases are used (in particular, orthonormal bases). While we will discuss compression in detail in Chapter 13, here, we just discuss its main theme: a small number of transform coefficients should capture a large part of the energy of the original sequence. In audio compression, the following characteristics are important:

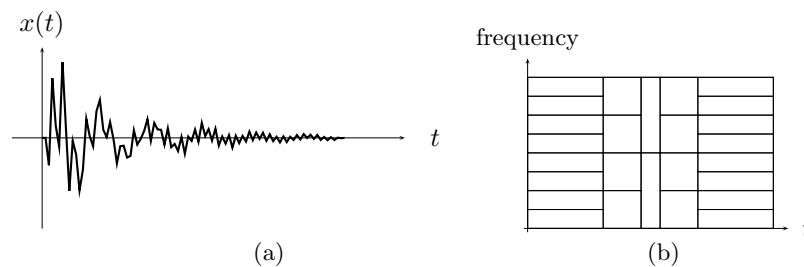
- (i) The spectrum is often harmonic, with a few dominant spectral components.
- (ii) The human auditory system exhibits a masking effect such that a large sinusoid masks neighboring smaller sinusoids.
- (iii) Sharp transitions, or attacks, are a key feature of many instruments.

It is clear that (i) and (iii) are in contradiction. The former requires long prototype windows, with local frequency analysis, while the latter requires short prototype windows, with global frequency analysis. The solution is to adapt the prototype window size, depending on the sequence content.

<sup>121</sup>Audio coding schemes use this feature extensively, as it allows for switching the number of channels in a filter bank, and consequently, the time and frequency resolutions of the analysis.



**Figure 8.16:** An example of the flexibility allowed by LOTs illustrated through different transitions from an 2-channel LOT to an 8-channel LOT. (a) Direct transition (both prototype windows have the same tails, a restriction on the 8-channel LOT as its prototype window must then be flat in the middle). (b) Transition using an asymmetric 4-channel LOT prototype window (allows for a greater flexibility in the 8-channel LOT prototype window). (c) Transition using an asymmetric 4-channel LOT prototype window and a symmetric 4-channel LOT prototype window. (d) Transition using several symmetric 4-channel LOT prototype windows (all the prototype windows are now symmetric and have the same tails).

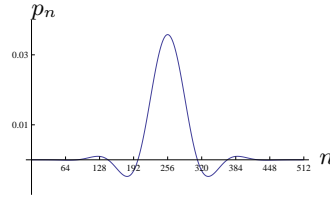


**Figure 8.17:** Analysis of an audio segment using a cosine-modulated filter bank. (a) Time-domain sequence. (b) Tiling of the time-frequency plane where shading indicates the square of the coefficient corresponding to basis sequence situated at that specific time-frequency location.

Both for harmonic analysis and for windowing, including changing the size of the filter bank (we have just seen this), we use cosine-modulated filter banks similar to those from Section 8.4, creating an adaptive tiling of the time-frequency plane, with local frequency resolution in stationary, harmonic segments, and local time resolution in transition, or, attack phases. The best tiling is chosen based on optimization procedures that try to minimize the approximation error when keeping only a small number of transform coefficients (we discuss such methods in Chapter 13). Figure 8.17 gives an example of adaptive time-frequency analysis.

For actual compression, in addition to an adaptive representation, a number of other tricks come into play, related to perceptual coding (for example, masking), quantization and entropy coding, all specifically tuned to audio compression.<sup>122</sup>

<sup>122</sup>All of this is typically done off line, that is, on the recorded audio, rather than in real time. This allows for complex optimizations, potentially using trial and error, until a satisfactory solution



**Figure 8.18:** Impulse response of the prototype window sequence modulating the cosine-modulated filter bank used in MP3.

**EXAMPLE 8.5 (FILTER BANKS USED IN AUDIO COMPRESSION)** The MPEG audio standard<sup>123</sup>, often called MP3 in consumer products, uses a 32-channel filter bank. It is not a perfect reconstruction filter bank; rather, it uses a symmetric prototype window  $p_n$  of length  $L = 2N - 1$  (in this case  $L = 511$ ) with a symmetry around  $n = 255$ . The  $i$ th filter is obtained by modulation of the prototype window as

$$g_{i,n} = p_n \cos \left( \frac{2\pi}{2N} \left( i + \frac{1}{2} \right) \left( n + \frac{N}{2} \right) \right), \quad (8.43)$$

for  $i = 0, 1, \dots, N-1$ . Comparing this to (8.30), we see that, except for the phase factor, the cosine modulation is the same. Of course, the prototype window is also different (it is of odd length hinting at the phase difference). The impulse response of the prototype window used in MP3 is displayed in Figure 8.18. Such a filter bank is called *pseudo-QMF*, because nearest neighbor aliasing is canceled as in a classical two-channel filter bank.<sup>124</sup> While aliasing from other, further bands, is not automatically canceled, the prototype is a very good lowpass suppressing it almost perfectly. The input-output relationship is not perfect (unlike for LOTs), but again, with a good prototype window, it is almost perfect.

## 8.5 Computational Aspects

The expressions for the synthesis and analysis complex exponential-modulated filter banks in (8.19a) and (8.19b) (see an illustration with three channels in Figure 8.4) lead to the corresponding fast algorithms given in Tables 8.3 and 8.4.

**Complex Exponential-Modulated Filter Banks** We now look into the cost of implementing the analysis filter bank; the cost of implementing the synthesis one is the same, as the two are dual to each other. Consider a prototype filter  $p$  of length  $L = NM$ ; each polyphase component is then of length  $N$ .

is obtained.

<sup>123</sup>While MPEG is a video standardization body, MP3 is its subgroup dealing with audio. Several different versions of audio compression, of different complexity and quality, have been developed, and the best of these, called layer III, gave the acronym MP3.

<sup>124</sup>The QMF filters are discussed in *Further Reading* of Chapter 7, as well as Exercise 7.19.

---

**ComplexModSynthesis**( $p, \{\alpha_0, \dots, \alpha_{N-1}\}$ )

**Input:** The prototype filter  $p$  and  $N$  channel sequences  $\{\alpha_0, \dots, \alpha_{N-1}\}$ .

**Output:** Original sequence  $x$ .

---

 Decompose prototype  $p$  into its  $N$  polyphase components  $p_{j,n} = p_{Nn+j}$ 
**for all**  $n$  **do**

   *Fourier transform:* Transform channel sequences with the scaled inverse DFT

$$\begin{bmatrix} \alpha'_{0,n} \\ \alpha'_{1,n} \\ \vdots \\ \alpha'_{N-1,n} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_N^{-1} & W_N^{-2} & \dots & W_N^{-(N-1)} \\ 1 & W_N^{-2} & W_N^{-4} & \dots & W_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{-(N-1)} & W_N^{-2(N-1)} & \dots & W_N^{-(N-1)^2} \end{bmatrix} \begin{bmatrix} \alpha_{0,n} \\ \alpha_{1,n} \\ \vdots \\ \alpha_{N-1,n} \end{bmatrix}$$

*Convolution:* Convolve each sequence  $\alpha'_{j,n}$  with the  $j$ th polyphase component of  $p$ 

$$\begin{bmatrix} \alpha''_{0,n} \\ \alpha''_{1,n} \\ \vdots \\ \alpha''_{N-1,n} \end{bmatrix} = \begin{bmatrix} p_{0,n} & & & \\ & p_{1,n} & & \\ & & \ddots & \\ & & & p_{N-1,n} \end{bmatrix} * \begin{bmatrix} \alpha'_{0,n} \\ \alpha'_{1,n} \\ \vdots \\ \alpha'_{N-1,n} \end{bmatrix}$$

*Inverse polyphase transform:* Upsample/interleave channel sequences to get  $x_{Nn+j} = \alpha''_{j,n}$   
**end for**  
**return**  $x$ 
**Table 8.3:** Fast implementation of a complex exponential-modulated synthesis filter bank.

First, we need to compute  $M$  convolutions, but on polyphase components of the input sequence, that is, at a rate  $M$  times slower. This is equivalent to a single convolution at full rate, or, of order  $O(N)$  operations per input sample. We then use an FFT, again at the slower rate. From (2.261), an FFT requires of the order  $O(\log_2 M)$  operations per input sample. In total, we have

$$C \sim \alpha \log_2 M + N \sim O(\log_2 M), \quad (8.44)$$

operations per input sample. This is very efficient, since simply taking a length- $M$  FFT for each consecutive block of  $M$  samples would already require  $\log_2 M$  operations per input sample. Thus, the price of windowing given by the prototype filter is of the order  $O(N)$  operations per input sample, or, the length of the prototype window normalized per input sample. A value for  $N$  depends on the desired frequency selectivity; a typical value can be of order  $O(\log_2 M)$ . Exercise 8.4 looks into the cost of a filter bank similar to those used in audio compression standards, such as MPEG from Example 8.5.

What is the numerical conditioning of this algorithm? Clearly, both the polyphase transform and the FFT are unitary maps, so the key resides in the diagonal matrix of polyphase components. While there are cases when it is unitary (such as in the block-transform case where it is the identity), it is highly dependent on the prototype window. See Exercise 8.5 for an exploration of this issue.

**ComplexModAnalysis**( $p, x$ )**Input:** The prototype filter  $p$  and input  $x$ .**Output:**  $N$  channel sequences  $\{\alpha_0, \dots, \alpha_{N-1}\}$ .Decompose prototype  $p$  into its  $N$  polyphase components  $p_{j,n} = p_{Nn-j}$ **for all**  $n$  **do***Polyphase transform:* Compute input sequence polyphase components  $x_{j,n} = x_{Nn+j}$ *Convolution:* Convolve polyphase components of prototype and input

$$\begin{bmatrix} \alpha'_{0,n} \\ \alpha'_{1,n} \\ \vdots \\ \alpha'_{N-1,n} \end{bmatrix} = \begin{bmatrix} p_{0,n} & & & \\ & p_{1,n} & & \\ & & \ddots & \\ & & & p_{N-1,n} \end{bmatrix} * \begin{bmatrix} x_{0,n} \\ x_{1,n} \\ \vdots \\ x_{N-1,n} \end{bmatrix}$$

*Fourier transform:* Compute channel sequences by applying the forward DFT

$$\begin{bmatrix} \alpha_{0,n} \\ \alpha_{1,n} \\ \vdots \\ \alpha_{N-1,n} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_N & W_N^2 & \dots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \dots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{(N-1)} & W_N^{2(N-1)} & \dots & W_N^{(N-1)^2} \end{bmatrix} \begin{bmatrix} \alpha'_{0,n} \\ \alpha'_{1,n} \\ \vdots \\ \alpha'_{N-1,n} \end{bmatrix}$$

**end for****return**  $\{\alpha_0, \dots, \alpha_{N-1}\}$ **Table 8.4:** Fast implementation of a complex exponential-modulated analysis filter bank.**Chapter at a Glance**

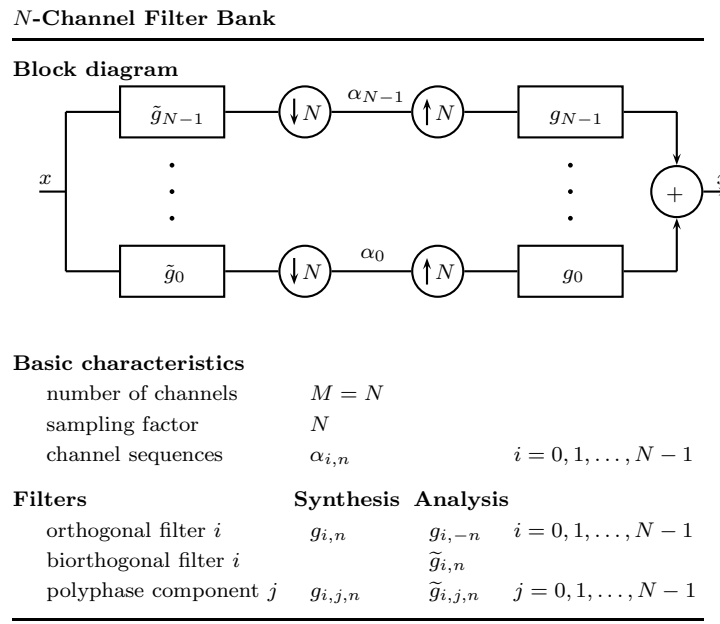
Our goal in this chapter was twofold: (1) to extend the discussion from Chapter 7 to more than two channels and associated bases; and (2) to consider those filter banks implementing local Fourier bases.

The extension to  $N$  channels, while not difficult, is a bit more involved as we now deal with more general matrices, and, in particular,  $N \times N$  matrices of polynomials. Many of the expressions are analogous to those seen Chapter 7; we went through them in some detail for orthogonal  $N$ -channel filter banks, as the biorthogonal ones are similar.

General, unstructured  $N$ -channel filter banks are rarely seen in practice; instead,  $N$ -channel modulated filter banks are widespread because (1) of their close connection to local Fourier representations, (2) computational efficiency, (modulated filter banks are implemented using FFTs), and (3) only a single prototype filter needs to be designed.

We studied uniformly-modulated filters bank using both complex exponentials and as well as cosines. The former, while directly linked to a local Fourier series (indeed, when the filter length is  $N$ , we have a blockwise DFT), is hampered by a negative result, Balian-Low theorem, which prohibits good orthonormal bases. The latter, with proper design of the prototype filter, leads to good, orthonormal, local cosine bases (LOTs). These are popular in audio and image processing using a prototype filter of length  $L = 2N$ .

To showcase their utility, we looked at the use of complex exponential-modulated filter banks in power spectral density estimation, communications (OFDM) and transmultiplexing (Wi-Fi), as well as that of cosine-modulated ones in audio compression (MP3).

**Table 8.5:**  $N$ -channel filter bank.

Local Fourier Modulated Filter Bank			
Filters	Modulation		
	complex-exponential	cosine	
$g_{i,n}$	$p_n W_N^{-in}$	$p_n \cos\left(\frac{2\pi}{2N}\left(i + \frac{1}{2}\right)n + \theta_i\right)$ $p_n \frac{1}{2} \left[ e^{j\theta_i} W_{2N}^{-(i+1/2)n} + e^{-j\theta_i} W_{2N}^{(i+1/2)n} \right]$	
$G_i(z)$	$P(W_N^i z)$	$\frac{1}{2} \left[ e^{j\theta_i} P(W_{2N}^{(i+1/2)} z) + e^{-j\theta_i} P(W_{2N}^{-(i+1/2)} z) \right]$	
$G_i(e^{j\omega})$	$P(e^{j(\omega - (2\pi/N)i)})$	$\frac{1}{2} \left[ e^{j\theta_i} P(e^{j(\omega - (2\pi/2N)(i+1/2))}) + e^{-j\theta_i} P(e^{j(\omega + (2\pi/2N)(i+1/2))}) \right]$	

**Table 8.6:** Local Fourier bases with complex and cosine modulation.



<b><i>N</i>-Channel Orthogonal Filter Bank</b>		
<b>Relationship between filters</b>		
Time domain	$\langle g_{i,n}, g_{j,n-Nk} \rangle_n = \delta_{i-j} \delta_k$	
Matrix domain	$D_N G_j^T G_i U_N = \delta_{i-j}$	
$z$ domain	$\sum_{k=0}^{N-1} G_i(W_N^k z) G_j(W_N^{-k} z^{-1}) = N \delta_{i-j}$	
DTFT domain	$\sum_{k=0}^{N-1} G_i(e^{j(\omega - (2\pi/N)k)}) G_j(W_N^k e^{-j\omega}) = N \delta_{i-j}$	
Polyphase domain	$\sum_{k=0}^{N-1} G_{i,k}(z) G_{j,k}(z) = \delta_{i-j}$	
<b>Basis sequences</b>	<b>Time domain</b>	<b>Frequency domain</b>
	$\{g_{i,n-2k}\}_{i=0,\dots,N-1,k \in \mathbb{Z}}$	$\{G_i(z)\}_{i=0,\dots,N-1}$
<b>Filters</b>	<b>Synthesis</b>	<b>Analysis</b>
	$g_{i,n}, G_i(z), G_i(e^{j\omega})$	$g_{i,-n}, G_i(z^{-1}), G_i(e^{-j\omega})$
<b>Matrix view</b>	<b>Basis</b>	
Time domain	$\Phi$	$\begin{bmatrix} \dots & g_{0,n-2k} & g_{1,n-2k} & \dots & g_{0N-1,n-2k} \end{bmatrix}$
$z$ domain	$\Phi(z)$	$\begin{bmatrix} G_0(z) & G_1(z) & \dots & G_{N-1}(z) \\ G_0(W_N z) & G_1(W_N z) & \dots & G_{N-1}(W_N z) \\ \vdots & \vdots & \ddots & \vdots \\ G_0(W_N^{N-1} z) & G_1(W_N^{N-1} z) & \dots & G_{N-1}(W_N^{N-1} z) \end{bmatrix}$
DTFT domain	$\Phi(e^{j\omega})$	$\begin{bmatrix} G_0(e^{j\omega}) & G_1(e^{j\omega}) & \dots & G_{N-1}(e^{j\omega}) \\ G_0(W_N e^{j\omega}) & G_1(W_N e^{j\omega}) & \dots & G_{N-1}(W_N e^{j\omega}) \\ \vdots & \vdots & \ddots & \vdots \\ G_0(W_N^{N-1} e^{j\omega}) & G_1(W_N^{N-1} e^{j\omega}) & \dots & G_{N-1}(W_N^{N-1} e^{j\omega}) \end{bmatrix}$
Polyphase domain	$\Phi_p(z)$	$\begin{bmatrix} G_{0,0}(z) & G_{1,0}(z) & \dots & G_{N-1,0}(z) \\ G_{0,1}(z) & G_{1,1}(z) & \dots & G_{N-1,1}(z) \\ \vdots & \vdots & \ddots & \vdots \\ G_{0,N-1}(z) & G_{1,N-1}(z) & \dots & G_{N-1,N-1}(z) \end{bmatrix}$
<b>Constraints</b>	<b>Orthogonality relations</b>	<b>Perfect reconstruction</b>
Time domain	$\Phi^* \Phi = I$	$\Phi \Phi^* = I$
$z$ domain	$\Phi(z^{-1})^* \Phi(z) = I$	$\Phi(z) \Phi^*(z^{-1}) = I$
DTFT domain	$\Phi^*(e^{-j\omega}) \Phi(e^{j\omega}) = I$	$\Phi(e^{j\omega}) \Phi^*(e^{-j\omega}) = I$
Polyphase domain	$\Phi_p^*(z^{-1}) \Phi_p(z) = I$	$\Phi_p(z) \Phi_p^*(z^{-1}) = I$

**Table 8.7:** Properties of an orthogonal  $N$ -channel filter bank. This table is the  $N$ -channel counterpart to the two-channel Table 7.9.

## Historical Remarks

The earliest application of a local Fourier analysis was by Dennis Gabor to the analysis of speech [55]. The idea of a local spectrum, or periodogram, was studied and refined by statisticians interested in time series of sun spots, floods, temperatures, and many others. It led to the question of windowing the data. Blackman, Tukey, Hamming, among others, worked on window designs, while the question of smoothing was studied by Bartlett and Welch, producing windowed and smoothed periodograms.



**MP3** For compression, especially speech and audio, real modulated local Fourier filter banks with perfect or almost perfect reconstruction appeared in the 1980's and 1990's. Nussbaumer, Rothweiler and others proposed pseudo-QMF filter banks, with nearly perfect reconstruction, frequency selective filters and high computational efficiency. This type of filter bank is used today

in most audio coding standards, such as MP3. A different approach, leading to shorter filters and LOTs, was championed by Malvar, Princen and Bradley, among others. These are popular in image processing, where frequency selectivity is not as much of a concern.

**Wi-Fi** Frequency division multiplexing has been a popular communications method since the 1960's, and its digital version led to complex exponential-modulated transmultiplexers with FFTs, as proposed by Bellanger and co-workers. That perfect transmultiplexing is possible was pointed out by Vetterli. Multicarrier frequency signaling, which relies on efficient complex exponential-modulated transmultiplexers is one of the main communications methods, with orthogonal frequency division multiplexing (OFDM) being at the heart of many standards (for example, Wi-Fi, 802.11).



## Further Reading

**Books and Textbooks** For a general treatment of  $N$ -channel filter banks, see the books by Vaidyanathan [158], Vetterli and Kovačević [167], Strang and Nguyen [143], among others. For modulated filter banks, see [158] as well as Malvar's book [101], the latter with a particular emphasis on LOTs. For a good basic discussion of periodograms, see Porat's book on signal processing [114], while a more advanced treatment of spectral estimation can be found in Porat's book on statistical signal processing [113], and Stoica and Moses' book on spectral estimation [138].

**Design of  $N$ -Channel Filter Banks** General  $N$ -channel filter bank designs were investigated in [156, 157]. The freedom in design offered by more channels shows in examples such as linear-phase, orthogonal FIR solutions, not possible in the two-channel case [168, 136].

## Exercises with Solutions

### 8.1. Orthogonal Bases for $\ell^2(\mathbb{Z})$

Let  $C_3$  be the set of sequences in  $\ell^2(\mathbb{Z})$  that are constant over intervals of size 3, that is, a sequence  $x_n$  belongs to  $C_3$  if

$$x_{3k} = x_{3k+1} = x_{3k+2},$$

for all  $k \in \mathbb{Z}$ .

- (i) Prove that  $C_3$  is a subspace of  $\ell^2(\mathbb{Z})$ .
- (ii) Find an orthonormal basis for  $C_3$ , that is, describe the set of basis vectors, prove that the set is orthonormal and prove that any element in  $C_3$  can be written in terms of those vectors.
- (iii) Using filtering, upsampling and downsampling, show how to implement an orthogonal projection from  $\ell^2(\mathbb{Z})$  to  $C_3$ .
- (iv) Construct an orthogonal filter bank such that the projection operator from the previous part corresponds to one of the branches of the analysis/synthesis banks.

*Solution:*

- (i) To prove that  $C_3$  is a subspace we need to prove the following:
  - (a) If  $x, y$  are in  $C_3$ , then  $z = x + y$  is in  $C_3$  as well. For  $i = 0, 1, 2$ ,

$$z_{3k} = x_{3k} + y_{3k} = x_{3k+i} + y_{3k+i} = z_{3k+i}.$$

- (b) If  $x$  is in  $C_3$ , then  $z = \alpha x$ , where  $\alpha$  is a real or a complex number, is in  $C_3$  as well. For  $i = 0, 1, 2$ ,

$$z_{3k} = \alpha x_{3k} = \alpha x_{3k+i} = z_{3k+i}.$$

- (ii) Define the set of basis vectors as  $\{\varphi_{k,n} = \varphi_{n-3k} \mid n, k \in \mathbb{Z}\}$ , with  $\varphi_n = 1/\sqrt{3}$  for  $n = 0, 1, 2$ , and 0 otherwise.

- (a) To prove that the set is orthonormal, write

$$\langle \varphi_{k,n}, \varphi_{l,n} \rangle = \sum_n \varphi_{k,n} \varphi_{l,n} = \sum_n \varphi_{n-3k} \varphi_{n-3l}.$$

In the previous sum,  $\varphi_{n-3k}$  is nonzero for  $n = 3k, 3k+1, 3k+2$ , while  $\varphi_{n-3l}$  is nonzero for  $n = 3l, 3l+1, 3l+2$ . Thus

$$\langle \varphi_{k,n}, \varphi_{l,n} \rangle = \delta_{k-l}.$$

- (b) To prove completeness, we have to show that any element in  $C_3$  can be written as

$$x_n = \sum_k \langle \varphi_{k,l}, x_l \rangle \varphi_{k,n}.$$

The transform coefficients are given by

$$\langle \varphi_{k,l}, x_l \rangle = \frac{1}{\sqrt{3}}(x_{3k} + x_{3k+1} + x_{3k+2}).$$

Write now  $n = 3m + i$ , with  $i = 0$ , or 1 or 2. Then

$$x_n = \frac{1}{\sqrt{3}} \sum_k (x_{3k} + x_{3k+1} + x_{3k+2}) \varphi_{3(m-k)+i}.$$

Now,  $\varphi_{3(m-k)+i}$  is nonzero only for  $q = 3(m-k) + i$  equal to 0, or 1 or 2. Since  $3(m-k) = q - i$ , we can check all possible combinations of  $q, i$  and just take those that for  $q - i$  give an integer multiple of 3. This leads to  $q = i$ , or  $k = m$ , and thus

$$x_n = \frac{1}{\sqrt{3}}(x_{3m} + x_{3m+1} + x_{3m+2}) \frac{1}{\sqrt{3}}.$$

Since one of the elements  $x_{3m}$  or  $x_{3m+1}$  or  $x_{3m+2}$  is equal to  $x_n$ , and  $x_n$  belongs to  $C_3$ , that means that  $x_n = x_{3m} = x_{3m+1} = x_{3m+2}$  and thus

$$x_n = \frac{1}{3} \cdot 3 \cdot x_n = x_n.$$

- (iii) We know that the orthogonal projection is

$$\hat{x}_n = \sum_k \langle \psi_{k,l}, x_l \rangle \psi_{k,n},$$

where  $\{\psi_{k,l}\}$  is an orthonormal basis for  $C_3$ . We use the orthonormal basis for  $C_3$  we just constructed. That is, choose  $\psi_k = \varphi_k$  as our basis functions. We can check that the output  $\hat{x}_n$  is indeed in  $C_3$ . For  $i = 0, 1, 2$ ,

$$\hat{x}_{3m+i} = \frac{1}{3}(x_{3m} + x_{3m+1} + x_{3m+2}) = \frac{1}{3}3 \cdot x_{3m} = x_{3m}.$$

If we denote  $g_n = (\delta_n + \delta_{n-1} + \delta_{n-2})/\sqrt{3}$ , we can easily implement this orthogonal projection operator as filtering by  $g_{-n}$ , downsampling by 3, upsampling by 3 and filtering by  $g_n$ .

- (iv) We construct a 3-channel filter bank with sampling by 3 where the synthesis lowpass filter is given by
- $g_{0,n} = \varphi_n = 1/\sqrt{3}$
- for
- $n = 0, 1, 2$
- and 0 otherwise. Thus, we know the first row of the analysis polyphase matrix. We can use the simplest 3-channel orthogonal matrix given (8.15a),

$$\begin{aligned} \Phi_p(z) &= \begin{bmatrix} \cos \alpha_1 & 0 & \sin \alpha_1 \\ 0 & 1 & 0 \\ -\sin \alpha_1 & 0 & \cos \alpha_1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha_2 & \sin \alpha_2 \\ 0 & -\sin \alpha_2 & \cos \alpha_2 \end{bmatrix} \\ &= \begin{bmatrix} \cos \alpha_1 & -\sin \alpha_1 \sin \alpha_2 & \sin \alpha_1 \cos \alpha_2 \\ 0 & \cos \alpha_2 & \sin \alpha_2 \\ -\sin \alpha_1 & -\cos \alpha_1 \sin \alpha_2 & \cos \alpha_1 \cos \alpha_2 \end{bmatrix}. \end{aligned}$$

Since we know the first row, we know that  $\cos \alpha_1 = 1/\sqrt{3}$ . Therefore,  $\sin \alpha_1 = \sqrt{2}/\sqrt{3}$ . From the second element in the first row, we get that  $-\sin \alpha_1 \sin \alpha_2 = 1/\sqrt{3}$ . Thus,  $\sin \alpha_2 = -1/\sqrt{2}$  and finally  $\cos \alpha_2 = \pm 1/\sqrt{2}$ . Choose a “+” sign. The final polyphase matrix is

$$\Phi_p(z) = \begin{bmatrix} 1/\sqrt{3} & 1/\sqrt{3} & 1/\sqrt{3} \\ 0 & 1/\sqrt{2} & -1/\sqrt{2} \\ -\sqrt{2}/\sqrt{3} & 1/\sqrt{6} & 1/\sqrt{6} \end{bmatrix}.$$

It is easy to check that  $\Phi_p(z)\Phi_p^T(z^{-1}) = I$ .

### 8.2. Factorization of Complex Exponential-Modulated Modulation Matrices

Given is a 3-channel complex exponential-modulated local Fourier filter bank as in (8.16). We call the following circulant matrix:

$$\Phi_m(z) = \begin{bmatrix} G_0(z) & G_1(z) & G_2(z) \\ G_0(W_3z) & G_1(W_3z) & G_2(W_3z) \\ G_0(W_3^2z) & G_1(W_3^2z) & G_2(W_3^2z) \end{bmatrix} = \begin{bmatrix} G(z) & G(W_3z) & G(W_3^2z) \\ G(W_3z) & G(W_3^2z) & G(z) \\ G(W_3^2z) & G(z) & G(W_3z) \end{bmatrix},$$

the modulation matrix.

- Find the relationship between the modulation matrix  $\Phi_m(z)$  and the polyphase matrix  $\Phi_p(z)$ .
- Show how to diagonalize  $\Phi_m(z)$ .
- Give a form of the determinant  $\det(\Phi_m(z))$ .

*Solution:* The gist of this problem is to solve (i), that is, find the relationship between the polyphase and modulation matrices.

- We start with  $N = 2$ , as it is easier to see the relationship between the polyphase and modulation matrices in that case:

$$\begin{aligned} X(z) + X(-z) &= 2 \sum_{n \in \mathbb{Z}} x_{2n} z^{-n} = 2X_0(z^2), \\ X(z) - X(-z) &= 2z^{-1} \sum_{n \in \mathbb{Z}} x_{2n+1} z^{-n} = 2z^{-1}X_1(z^2), \end{aligned}$$

which can be compactly represented as

$$X_p(z^2) = \begin{bmatrix} X_0(z) \\ X_1(z) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & \\ & z \end{bmatrix} \underbrace{\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}}_F \underbrace{\begin{bmatrix} X(z) \\ X(-z) \end{bmatrix}}_{X_m(z)} = \frac{1}{2} \begin{bmatrix} 1 & \\ & z \end{bmatrix} F X_m(z).$$

In other words, the modulation and polyphase domain representations are related via the DFT. We can easily generalize this to arbitrary  $N$  to obtain:

$$X_p(z^N) = \frac{1}{N} \text{diag}([1 \quad z \quad \dots \quad z^{N-1}]) F X_m(z), \quad (\text{E8.2-1a})$$

$$\Phi_p(z^N) = \frac{1}{N} \text{diag}([1 \quad z \quad \dots \quad z^{N-1}]) F \Phi_m(z), \quad (\text{E8.2-1b})$$

where  $F$  is the DFT matrix from (2.161a).

- (ii) Since the second and third filters are modulates of the first one, we can rewrite the polyphase matrix as

$$\begin{aligned} \Phi_p(z) &= \begin{bmatrix} G_{0,0}(z) & G_{1,0}(z) & G_{2,0}(z) \\ G_{0,1}(z) & G_{1,1}(z) & G_{2,1}(z) \\ G_{0,2}(z) & G_{1,2}(z) & G_{2,2}(z) \end{bmatrix} \stackrel{(a)}{=} \begin{bmatrix} P_0(z) & P_0(z) & P_0(z) \\ P_1(z) & W^2 P_1(z) & W P_1(z) \\ P_2(z) & W P_2(z) & W^2 P_2(z) \end{bmatrix} \\ &= \text{diag}([P_0(z) \quad P_1(z) \quad P_2(z)]) F^*, \end{aligned}$$

where (a) follows from (8.11) (this was also derived in (8.17)). Using this and (E8.2-1b), we have that

$$\text{diag}([P_0(z^3) \quad P_1(z^3) \quad P_2(z^3)]) F^* = \frac{1}{3} \text{diag}([1 \quad z \quad z^2]) F \Phi_m(z),$$

and

$$\frac{1}{3} \text{diag}([1 \quad z \quad z^2]) F \Phi_m(z) (F^*)^{-1} = \text{diag}([P_0(z^3) \quad P_1(z^3) \quad P_2(z^3)]),$$

or, finally

$$F \Phi_m(z) F = 9 \text{diag}([P_0(z^3) \quad z^{-1} P_1(z^3) \quad z^{-2} P_2(z^3)]),$$

where we have used the expression for the adjoint of  $F$  in (2.161b).

- (iii) Using the factorization from (ii),

$$\begin{aligned} \det(\Phi_m(z)) &= \det(9 F^{-1} \text{diag}([P_0(z^3) \quad z^{-1} P_1(z^3) \quad z^{-2} P_2(z^3)]) F^{-1}) \\ &= 9^3 \det(F^{-1})^2 z^{-3} \prod_{j=0}^2 P_j(z^3) \stackrel{(a)}{=} 9^3 (\sqrt{3})^2 z^{-3} \prod_{j=0}^2 P_j(z^3) \\ &= 3^7 z^{-3} \prod_{j=0}^2 P_j(z^3). \end{aligned}$$

where (a) follows from (2.162).

### 8.3. Discrete Cosine Transform

The DCT is a block transform, where each basis sequence is a modulated version of an  $N$ -point average. One reason for its popularity is that the DCT matrix can be recursively factored in a fashion similar to the DFT; it can also be computed with a DFT of the same length and simple pre- and post-processing.

There are eight versions of the DCT; here, we give the basis sequences of the most well-known one, DCT-2:

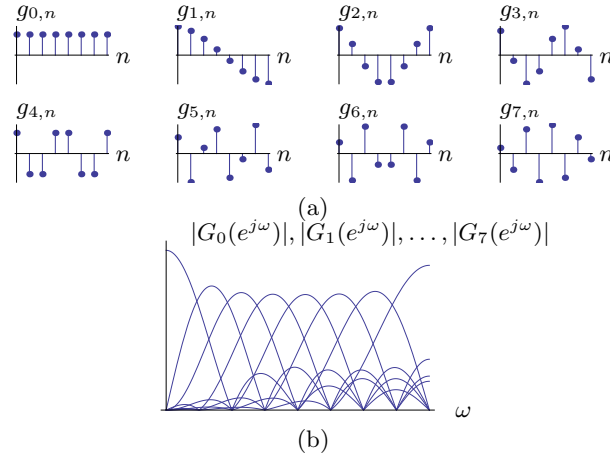
$$\varphi_{0,n} = g_{0,n} = p_n = \frac{1}{\sqrt{N}}, \quad (\text{E8.3-1a})$$

$$\varphi_{i,n} = g_{i,n} = p_n \sqrt{2} \cos\left(\frac{2\pi}{2N}\left(n + \frac{1}{2}\right)i\right), \quad (\text{E8.3-1b})$$

for  $i = 1, 2, \dots, N-1$ , and  $n = 0, 1, \dots, N-1$ . Prove that the DCT is an orthonormal basis for  $\mathbb{R}^N$ , by proving that:

$$\langle \varphi_i, \varphi_\ell \rangle = \delta_{i-\ell}.$$

Do this in steps:



**Figure E8.3-1:** DCT for  $N = 8$ . (a) The eight basis sequences. (b) Magnitude responses of the basis sequences, showing the uniform split of the spectrum.

- (i) Prove that  $\langle \varphi_i, \varphi_0 \rangle = 0$ , for  $i = 1, 2, \dots, N - 1$ .
- (ii) Prove that  $\langle \varphi_i, \varphi_\ell \rangle = 0$ , for  $i \neq \ell$ ,  $i, \ell = 1, 2, \dots, N - 1$ .
- (iii) Prove that  $\|\varphi_i\| = 1$  for  $i = 0, 1, \dots, N - 1$ .

For  $N = 8$ , numerically compare the DTFT magnitudes of the DCT basis sequences to those of the LOT basis sequences from (8.30).

*Solution:* The first three parts show that the DCT is an orthonormal basis for  $\mathbb{R}^N$ .

- (i) We show that all the basis sequences are orthogonal to the prototype one:

$$\begin{aligned}
 \langle \varphi_i, \varphi_0 \rangle &\stackrel{(a)}{=} \frac{\sqrt{2}}{N} \sum_{n=0}^{N-1} \cos\left(\frac{2\pi}{2N}i\left(n + \frac{1}{2}\right)\right) \\
 &\stackrel{(b)}{=} \frac{1}{\sqrt{2}N} \sum_{n=0}^{N-1} \left(W_{4N}^{i(2n+1)} + W_{4N}^{-i(2n+1)}\right) \\
 &= \frac{1}{\sqrt{2}N} \left(W_{4N}^i \sum_{n=0}^{N-1} W_{4N}^{2in} + W_{4N}^{-i} \sum_{n=0}^{N-1} W_{4N}^{-2in}\right) \\
 &\stackrel{(c)}{=} \frac{1}{\sqrt{2}N} \left(W_{4N}^i \frac{1 - W_{4N}^{2iN}}{1 - W_{4N}^{2i}} + W_{4N}^{-i} \frac{1 - W_{4N}^{-2iN}}{1 - W_{4N}^{-2i}}\right),
 \end{aligned}$$

where (a) follows from (E8.3-1); (b) from (2.275); and (c) from the finite-sum formula (P1.65-1). For  $i$  even,  $W_{4N}^{2iN} = 1$ , and the inner product is zero. For  $i$  odd,  $W_{4N}^{2iN} = -1$ , and the above becomes

$$\langle \varphi_i, \varphi_0 \rangle = \frac{\sqrt{2}}{N} \frac{W_{4N}^i (1 - W_{4N}^{-2i}) + W_{4N}^{-i} (1 - W_{4N}^{2i})}{(1 - W_{4N}^{2i})(1 - W_{4N}^{-2i})} = 0.$$

(ii) The argument follows a similar line as in (i), so we sketch it only:

$$\begin{aligned}
 \langle \varphi_i, \varphi_\ell \rangle &= \frac{2}{N} \sum_{n=0}^{N-1} \cos\left(\frac{2\pi}{2N}i\left(n + \frac{1}{2}\right)\right) \cos\left(\frac{2\pi}{2N}\ell\left(n + \frac{1}{2}\right)\right) \\
 &= \frac{1}{2N} \sum_{n=0}^{N-1} \left(W_{4N}^{i(2n+1)} + W_{4N}^{-i(2n+1)}\right) \left(W_{4N}^{\ell(2n+1)} + W_{4N}^{-\ell(2n+1)}\right) \\
 &= \frac{1}{2N} \left( W_{4N}^{(i+\ell)} \sum_{n=0}^{N-1} W_{4N}^{2(i+\ell)n} + W_{4N}^{-(i+\ell)} \sum_{n=0}^{N-1} W_{4N}^{-2(i+\ell)n} + \right. \\
 &\quad \left. W_{4N}^{(i-\ell)} \sum_{n=0}^{N-1} W_{4N}^{2(i-\ell)n} + W_{4N}^{-(i-\ell)} \sum_{n=0}^{N-1} W_{4N}^{-2(i-\ell)n} \right).
 \end{aligned}$$

The first two terms are equivalent to the problem solved in (i) with  $i' = i + \ell$ ; the same is true for the last two terms, just with  $i' = i - \ell$ , proving that the inner product is zero.

(iii) That  $\varphi_0$  is of unit norm is obvious. For  $\varphi_i$ ,  $i = 1, 2, \dots, N-1$ , we have that

$$\begin{aligned}
 \langle \varphi_i, \varphi_i \rangle &= \frac{1}{2N} \sum_{n=0}^{N-1} \left(W_{4N}^{i(2n+1)} + W_{4N}^{-i(2n+1)}\right)^2 \\
 \langle \varphi_i, \varphi_i \rangle &= \frac{1}{2N} \left( \sum_{n=0}^{N-1} W_{4N}^{2i(2n+1)} + \sum_{n=0}^{N-1} W_{4N}^{-2i(2n+1)} + 2 \sum_{n=0}^{N-1} W_{4N}^0 \right).
 \end{aligned}$$

The first two sums are zero by the same argument as in (i). The third sum equals  $N$ , showing that the norm of  $\varphi_i$  is indeed 1.

The DCT basis sequences as well as their magnitude responses are given in Figure E8.3-1. Compare this to their LOT counterpart in Figure 8.14.

- 8.4. *Complexity of a Cosine-Modulate Local Fourier Filter Bank for Audio Compression*  
 Like the MPEG filter bank in Example 8.5, a 32-channel filter bank with filters of length 512 as in (8.16) is used in a standard called MUSICAM.<sup>125</sup> This filter bank can be implemented using polyphase filters and an FFT. Assuming an input sampling rate of 44.1 kHz, give the number of operations per second required to compute the filter bank.

*Solution:* The complexity of computing the analysis filter bank is the same as that of computing the synthesis filter bank. We use (8.44) with  $M = 32$ ,  $L = 512$ , and thus  $K = M/L = 16$ . The number of operations per sample equals

$$2 \frac{L}{M} + 2 \log_2 M = 2 \frac{512}{32} + 2 \log_2 32 = 2(16 + 5) = 42.$$

With 44,100 samples per second, this amounts to 1,852,200 operations per second.

## Exercises

### 8.1. Projection Operator

Given is the system in Fig. P8.1-1 with  $\langle a_n, a_{n-2k} \rangle = \delta_k$ ,  $\langle b_n, b_{n-3k} \rangle = \delta_k$ .

- Is  $P$  in  $x_1 = Px$  a projection? Why? Prove it.
- Redraw the above block-diagram as below and give expressions for  $c_n, d_n, M$  and  $N$  in any domain convenient.

<sup>125</sup>Actually, in a real MUSICAM system, the modulation is with cosines and the implementation involves polyphase filters and a fast DCT, and is thus very similar to the complex case we analyze here.

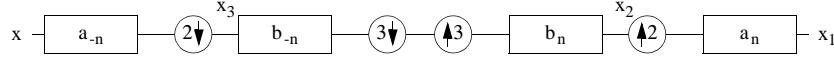
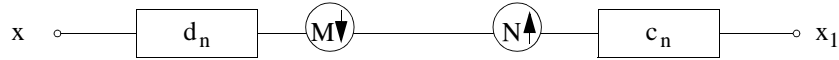


Figure P8.1-1: System for Exercise 8.1.



(iii) If  $a_n = (\delta_n + \delta_{n-1})/\sqrt{2}$  and  $b_n = (\delta_n + \delta_{n-1} + \delta_{n-2})/\sqrt{3}$ , what are  $c_n$  and  $d_n$ ?

(iv) For  $a_n$  and  $b_n$  as in (iii), does the following hold:

$$\langle c_n, c_{n-6k} \rangle = \delta_k? \quad (\text{P8.1-1})$$

(v) For general  $a_n$  and  $b_n$  as in (ii), does (P8.1-1) hold?

### 8.2. Biorthogonal Relations

Prove that the system shown in Figure P8.2-1 begin identity is equivalent to the following:

$$\langle g_{Nk-n}, \tilde{g}_{n-Nl} \rangle = \delta_{k-l}.$$

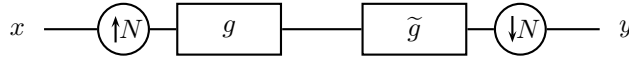


Figure P8.2-1: System for Exercise 8.2.

### 8.3. Biorthogonality in $N$ -Channel Perfect Reconstruction Filter Banks

Given is the modulation matrix  $\Phi_m(z)$ :

$$\Phi_m(z) = \begin{bmatrix} G_0(z) & G_1(z) & G_2(z) & \cdots & G_{N-1}(z) \\ G_0(Wz) & G_1(Wz) & G_2(Wz) & \cdots & G_{N-1}(Wz) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ G_0(W^{N-1}z) & G_1(W^{N-1}z) & G_2(W^{N-1}z) & \cdots & G_{N-1}(W^{N-1}z) \end{bmatrix}, \quad (\text{P8.3-1})$$

and similarly for the modulation matrix on the analysis side,  $\tilde{\Phi}_m(z)$ . Find the perfect reconstruction condition in terms of the modulation matrices  $\Phi_m$  and  $\tilde{\Phi}_m$ .

### 8.4. Complex Exponential-Modulated Local Fourier Basis with Ideal Filters

Consider an ideal  $N$ th band filter  $G$  from Table 2.5, and its modulations as in (8.16).

(i) Prove that the impulse responses and their shifts by multiples of  $N$ ,

$$\{g_{i,n-kN}\}_{k \in \mathbb{Z}, i \in \{0,1,\dots,N-1\}},$$

form an orthonormal set.

(ii) Prove that all filters are modulates of the prototype filter  $p$ , following (8.16), both in time and frequency domains.

### 8.5. Conditioning of Complex Exponential-Modulated Local Fourier Filter Banks

Given is a complex exponential-modulated local Fourier filter bank with filters  $G_i$  as in (8.16), and their polyphase components  $G_{i,j}$ ,  $i, j = 0, 1, \dots, N-1$ , as in (8.12d).



- (i) Take  $N = 2$ , where the two filters are  $G_0(z)$  and  $G_1(z) = G_0(-z)$ . Show that the conditioning of the filter bank is given by the conditioning of the matrix

$$M(e^{j\omega}) = \begin{bmatrix} |G_{0,0}(e^{j\omega})|^2 & 0 \\ 0 & |G_{0,1}(e^{j\omega})|^2 \end{bmatrix},$$

where, by conditioning, we mean that we want to find bounds  $\alpha$  and  $\beta$  such that

$$\alpha \|x\|^2 \leq \sum_{i=0}^{N-1} \|\alpha_i\|^2 \leq \beta \|x\|^2,$$

where the  $\alpha_i = \langle x, g_{i,n-2k} \rangle$  are the  $N$  channel signals.

(Hint: Take a fixed frequency  $\omega_0$  and find  $\alpha(\omega_0)$  and  $\beta(\omega_0)$ . Then extend the argument to  $\omega \in [-\pi, \pi)$ .)

- (ii) Compute the bounds  $\alpha$  and  $\beta$  for the following filters:

Filter	$G_0(z)$
Haar	$\frac{1}{\sqrt{2}}(1 + z^{-1})$
Ideal halfband	$\begin{cases} \sqrt{2}, &  \omega  \leq \pi/2; \\ 0, & \text{otherwise.} \end{cases}$
4-point average	$\frac{1}{4}(1 + z^{-1} + z^{-2} + z^{-3})$
Windowed average	$\sqrt{\frac{2}{5}}(\frac{1}{2} + z^{-1} + z^{-2} + \frac{1}{2}z^{-3})$

- (iii) Extend the argument to general  $N$ .  
 (iv) Numerically compute  $\alpha$  and  $\beta$  for the triangular windows below:

$N$	$G_0(z)$
4	$1 + 2z^{-1} + 3z^{-2} + 4z^{-3} + 4z^{-4} + 3z^{-5} + 2z^{-6} + z^{-7}$
3	$1 + 2z^{-1} + 3z^{-2} + 4z^{-3} + 5z^{-4} + 4z^{-5} + 3z^{-6} + 2z^{-7} + z^{-8}$

#### 8.6. Block-Based Power Spectral Density Estimation of White Noise

Consider a white noise process, or  $x_n$  is i.i.d. with variance  $\sigma_x^2$ .

- (i) Show that the entries of  $B_n$  in (8.21) are i.i.d. with variance  $\sigma_x^2$ , irrespective of the size  $M$  of the DFT.  
 (ii) Show that the average power spectrum (8.25) has variance  $(1/K)\sigma_x^2$ .

#### 8.7. Power Spectral Density Estimation Using Windowing

Consider windows  $p$  of odd size  $M$ , centered at the origin (such windows are not causal, but can be made so with a right shift by  $(M-1)/2$ ). For numerical evaluations and plots, use  $M = 31$ .

- (i) *Rectangular window*, or, the box sequence from (2.13a):

$$p_n^{(r)} = \begin{cases} 1/\sqrt{M}, & |n| \leq (M-1)/2; \\ 0, & \text{otherwise.} \end{cases} \quad (\text{P8.7-1})$$

Calculate its DTFT, the width of the main lobe (defined as the distance between zeroes of the DTFT around the origin) and the height of the tallest side lobe. Note that the DTFT of  $p^{(r)}$  is the Dirichlet kernel of order  $(M-1)/2$ .

- (ii) *Triangular (Bartlett) window*:

$$p_n^{(t)} = \begin{cases} 1 - (M+1)/2, & |n| \leq (M-1)/2; \\ 0, & \text{otherwise.} \end{cases} \quad (\text{P8.7-2})$$

Calculate its DTFT, the width of the main lobe and the height of the tallest side lobe.

(Hint: The triangular window can be seen as the convolution of a rectangular window of size  $(M+1)/2$  with itself.)

- (iii) Comparing the two windows above,  $p^{(r)}$  and  $p^{(t)}$ , show that, from (i) to (ii), the main lobe doubles while the side lobe is halved (in dB).
- (iv) *Hamming window*:

$$p_n^{(h)} = \begin{cases} 0.54 - 0.46 \cos\left(\frac{2\pi}{M-1}\left(n - \frac{M-1}{2}\right)\right), & |n| \leq (M-1)/2; \\ 0, & \text{otherwise.} \end{cases} \quad (\text{P8.7-3})$$

Calculate its DTFT, the width of the main lobe and the height of the tallest side lobe.

- (v) Verify that the Hamming window can be written in frequency domain as the sum of three Dirichlet kernels, one at the origin and weighted by 0.54, and the other two weighted by 0.23, shifted by  $\pm 2\pi/(M-1)$ . From this, give the expression for the width of the main lobe.

#### 8.8. Parametric Spectral Estimation

Consider signals consisting of complex sinusoids and additive white Gaussian noise. To detect a sinusoid, one takes a windowed DFT and looks for maxima in the squared magnitude.

- (i) Compare qualitatively the rectangular window  $p^{(r)}$  from (P8.7-1) and the triangular window  $p^{(t)}$  from (P8.7-2), for the following cases:
  - (i) Single complex sinusoid in low versus high noise.
  - (ii) Two closely spaced complex sinusoids in low versus high noise.
- (iii) Take  $M = 64$  and generate sequences with one or two sinusoids, and noise levels of the order of the largest side lobe of the rectangular and triangular windows  $p^{(r)}$  and  $p^{(t)}$  (see Problem 8.7). Compare the detection of sinusoids in these various cases.

#### 8.9. Transmultiplexers

For the transmultiplexer shown in Figure 8.10, verify the following for  $N = 3$  and  $N = 4$ :

- (i) If  $G_0(z) = \tilde{G}_0(z) = (1/\sqrt{N})(1 + z^{-1} + z^{-2} + \dots + z^{-N+1})$  and the filter bank is modulated as in (8.16), then the system is perfect reconstruction.
- (ii) If  $G_0(z) = \tilde{G}_0(z)$  are ideal  $N$ th-band filters and the filter bank is modulated as in (8.16), then the system is perfect reconstruction.
- (iii) If  $G_0(z)$  is of finite length (but larger than  $N$ ), and the filter bank is modulated as in (8.16), derive the input-output relationship in terms of the polyphase components of  $G_0(z)$  and  $\tilde{G}_0(z)$ .  
(Hint: Use the factorization analogous to the one in (8.17)).

## Chapter 9

# Wavelet Bases on Sequences

## Contents

9.1	Introduction . . . . .	676
9.2	Tree-Structured Filter Banks . . . . .	683
9.3	Orthogonal Discrete Wavelet Transform . . . . .	690
9.4	Biorthogonal Discrete Wavelet Transform . . . . .	696
9.5	Wavelet Packets . . . . .	698
9.6	Computational Aspects . . . . .	700
	Chapter at a Glance . . . . .	702
	Historical Remarks . . . . .	703
	Further Reading . . . . .	703
	Exercises with Solutions . . . . .	703
9.7	Introduction . . . . .	703
	Exercises . . . . .	710
9.8	Introduction . . . . .	710

If the projection of the signal onto two subspaces is advantageous, projecting onto more subspaces might be even better. These projections onto multiple subspaces are implemented via multichannel filter banks, which come in various flavors: For example, there are direct multichannel filter banks, with  $N$  filters covering the entire spectrum, their outputs downsampled by  $N$ , covered in Chapter 8. There are also tree-structured multichannel filter banks, where a two-channel filter bank from Chapter 7 is used as a building block for more complex structures. While we will discuss arbitrary tree structures later in this chapter, most of the chapter deals with a particularly simple one that has some distinguishing features, both from mathematical as well as practical points of view. This elementary tree structure recursively splits the coarse space into ever coarser ones, yielding, in signal processing parlance, an *octave-band* filter bank. The input spectrum (subspace) from 0 to  $\pi$  is cut into a highpass part from  $\pi/2$  to  $\pi$ , with the remainder cut again into  $\pi/4$  to  $\pi/2$  and a new remainder from 0 to  $\pi/4$ , and so on. As an example, performing

the split three times leads to the following 4-channel spectral division:

$$\left[0, \frac{\pi}{8}\right), \left[\frac{\pi}{8}, \frac{\pi}{4}\right), \left[\frac{\pi}{4}, \frac{\pi}{2}\right), \left[\frac{\pi}{2}, \pi\right),$$

yielding a lowpass (coarse) version and three bandpass (detail) versions, where each corresponds to an octave of the initial spectrum, shown in Figure 9.1(c).<sup>126</sup>

Such an unbalanced tree-structured filter bank shown in Figure 9.1 is a central concept both in filter banks as well as wavelets. Most of this chapter is devoted to its study, properties, and geometrical interpretation. In wavelet parlance, when the lowpass filter is designed appropriately, the filter bank computes a *discrete wavelet transform* (DWT). Even more is true: the same construction can be used to derive continuous-time wavelet bases, and the filter bank leads to an algorithm to compute wavelet series coefficients, the topic of Chapter 12.

## 9.1 Introduction

The iterated structure from Figure 9.1(a) clearly performs only the analysis operation. Given what we have learned so far, the channel sequences  $\beta^{(1)}$ ,  $\beta^{(2)}$ ,  $\beta^{(3)}$ ,  $\alpha^{(3)}$  compute projection coefficients onto some, yet unidentified, subspaces; it is left to establish which expansion has these as its projection/transform coefficients. Moreover, we should be able to then express the entire expansion using filter banks as we have done in Chapter 7. It is not difficult to see, that the synthesis filter bank corresponding to the analysis one from Figure 9.1(a), is the one in Figure 9.1(b). Every analysis two-channel block in Figure 9.1(a) has a corresponding synthesis two-channel block in Figure 9.1(b). We can thus use the whole machinery from Chapter 7 to study such an iterated structure. Moreover, the example with  $J = 3$  levels can be easily generalized to an arbitrary number of levels.

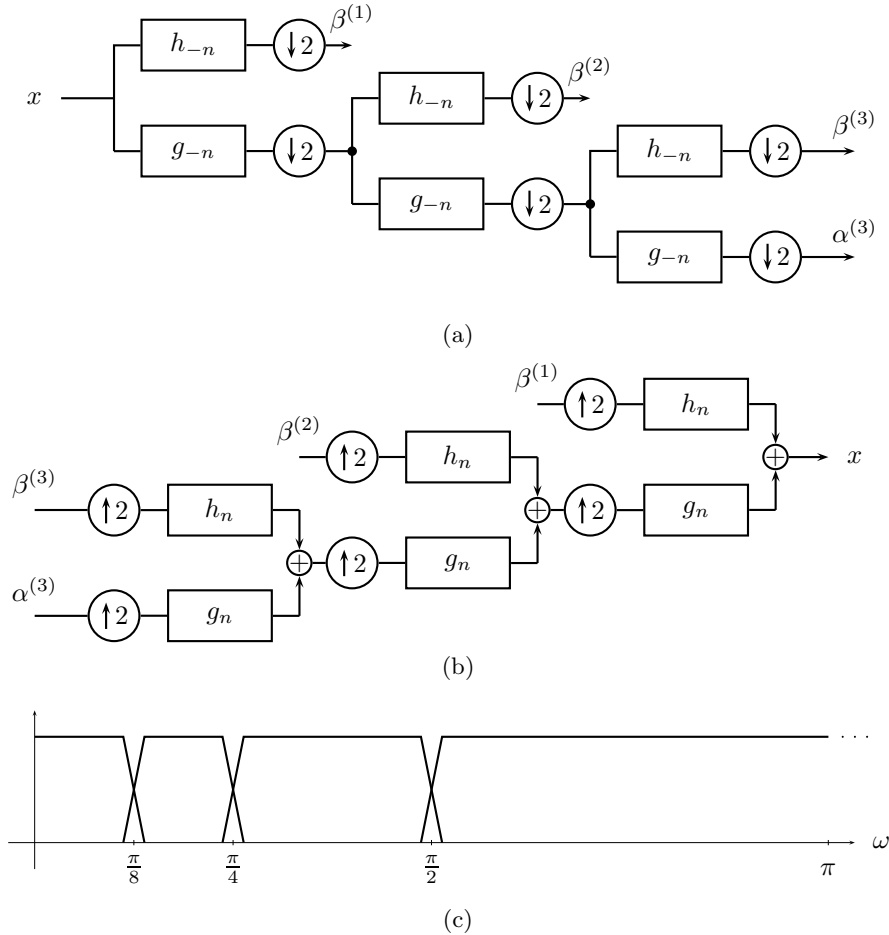
As we have done throughout the book, we introduce the main concepts of this chapter through our favorite example—Haar. Building upon the intuition we develop here, generalizations will come without surprise in the rest of the chapter.

### Implementing a Haar DWT Expansion

We start with a 3-level iterated filter bank structure as in Figure 9.1, where the two-channel filter bank block is the Haar orthogonal filter bank from Table 7.8, with synthesis filters

$$G(z) = \frac{1}{\sqrt{2}}(1 + z^{-1}), \quad H(z) = \frac{1}{\sqrt{2}}(1 - z^{-1}).$$

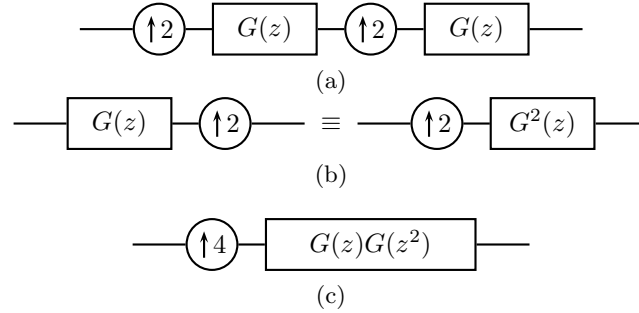
<sup>126</sup>Another interpretation of octave-band filter banks is that the bandpass channels have constant relative bandwidth. For a bandpass channel, its relative bandwidth  $Q$  is defined as its center frequency divided by its bandwidth. In the example above, the channels go from  $\pi/2^{i+1}$  to  $\pi/2^i$ , with the center frequency  $3\pi/2^{i+2}$  and bandwidth  $\pi/2^{i+1}$ . The relative bandwidth  $Q$  is then  $3/2$ . In classic circuit theory, the relative bandwidth is called the  $Q$ -factor, and the filter bank above has constant- $Q$  bandpass channels.



**Figure 9.1:** A two-channel orthogonal filter bank iterated three times to obtain one coarse subspace with support  $[0, \pi/8)$ , and three bandpass subspaces. (a) Analysis filter bank. (b) Synthesis filter bank. (c) The corresponding frequency division.

**Equivalent Filters** As mentioned earlier, we now have four channel sequences,  $\beta^{(1)}$ ,  $\beta^{(2)}$ ,  $\beta^{(3)}$ ,  $\alpha^{(3)}$ , and thus, we should be able to represent the tree structure from Figure 9.1 as a 4-channel filter bank, with four channel filters and four samplers. This is our aim now.

We first consider the channel sequence  $\alpha^{(3)}$  and its path through the lower branches of the first two filter banks, depicted in Figure 9.2(a). In part (b) of the same figure, we use one of the identities on the interchange of multirate operations and filtering we saw in Chapter 2, Figure 2.22, to move the first filter  $G(z)$  across the second upsampler, resulting in part (c) of the figure. In essence, we have compacted the sequence of steps “upsampler by 2—filter  $G(z)$ —upsampler by 2—filter  $G(z)$ ” into a sequence of steps “upsampler by 4—equivalent filter  $G^{(2)}(z) = G(z)G(z^2)$ ”.



**Figure 9.2:** Path through the lower branches of the first two filter banks in Figure 9.1. (a) Original system. (b) Use of one of the identities on the interchange of multirate operations and filtering from Figure 2.22 results in moving the filter across the upsampler by upsampling its impulse response. (c) Equivalent system consisting of a single upsampler by 4 followed by an equivalent filter  $G^{(2)}(z) = G(z)G(z^2)$ .

We can now iteratively continue the process by taking the equivalent filter and passing it across the third upsampler along the path of the lower branch in the last (rightmost) filter bank, resulting in a single branch with a single upsampler by 8 followed by a single equivalent filter  $G^{(3)}(z) = G(z)G(z^2)G(z^4)$ , resulting in the lowest branch of Figure 9.3.

Repeating the process on the other three branches transforms the 3-level tree-structured synthesis filter bank from Figure 9.1(b) into the 4-channel synthesis filter bank from Figure 9.3, with the equivalent filters:

$$H^{(1)}(z) = H(z) = \frac{1}{\sqrt{2}}(1 - z^{-1}), \quad (9.1a)$$

$$\begin{aligned} H^{(2)}(z) &= G(z)H(z^2) = \frac{1}{2}(1 + z^{-1})(1 - z^{-2}) \\ &= \frac{1}{2}(1 + z^{-1} - z^{-2} - z^{-3}), \end{aligned} \quad (9.1b)$$

$$\begin{aligned} H^{(3)}(z) &= G(z)G(z^2)H(z^4) = \frac{1}{2\sqrt{2}}(1 + z^{-1})(1 + z^{-2})(1 - z^{-4}) \\ &= \frac{1}{2\sqrt{2}}(1 + z^{-1} + z^{-2} + z^{-3} - z^{-4} - z^{-5} - z^{-6} - z^{-7}), \end{aligned} \quad (9.1c)$$

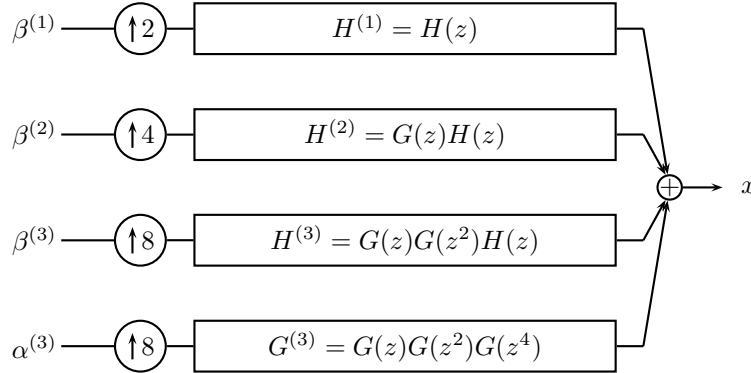
$$\begin{aligned} G^{(3)}(z) &= G(z)G(z^2)G(z^4) = \frac{1}{2\sqrt{2}}(1 + z^{-1})(1 + z^{-2})(1 + z^{-4}) \\ &= \frac{1}{2\sqrt{2}}(1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + z^{-5} + z^{-6} + z^{-7}). \end{aligned} \quad (9.1d)$$

If we repeated the above iterative process  $J$  times instead, the lowpass equivalent filter would have the  $z$ -transform

$$G^{(J)}(z) = \prod_{\ell=0}^{J-1} G(z^{2^\ell}) = \frac{1}{2^{J/2}} \sum_{n=0}^{2^J-1} z^{-n}, \quad (9.2a)$$

that is, it is a length- $2^J$  averaging filter

$$g_n^{(J)} = \frac{1}{2^{J/2}} \sum_{k=0}^{2^J-1} \delta_{n-k}, \quad (9.2b)$$



**Figure 9.3:** Equivalent filter bank to the 3-level synthesis bank shown in Figure 9.1(b).

while the same-level bandpass equivalent filter follows from

$$H^{(J)}(z) = H(z^{2^{J-1}})G^{(J-1)}(z) = \frac{1}{\sqrt{2}} \left( G^{(J-1)}(z) - z^{-2^{J-1}} G^{(J-1)}(z) \right), \quad (9.2c)$$

with the impulse response

$$h_n^{(J)} = \frac{1}{2^{J/2}} \left( \sum_{k=0}^{2^{J-1}-1} \delta_{n-k} - \sum_{k=2^{J-1}}^{2^J-1} \delta_{n-k} \right). \quad (9.2d)$$

**Basis Sequences** As we have done in Chapter 7, we now identify the resulting expansion and corresponding basis sequences. To each branch in Figure 9.3 corresponds a subspace spanned by the appropriate basis sequences. Let us start from the top. The first channel, with input  $\beta^{(1)}$ , has the equivalent filter  $h^{(1)} = h$ , just as in the basic two-channel filter bank, (7.10b), with upsampling by 2 in front. The corresponding sequences spanning the subspace  $W^{(1)}$  are (Figure 9.4(a)):

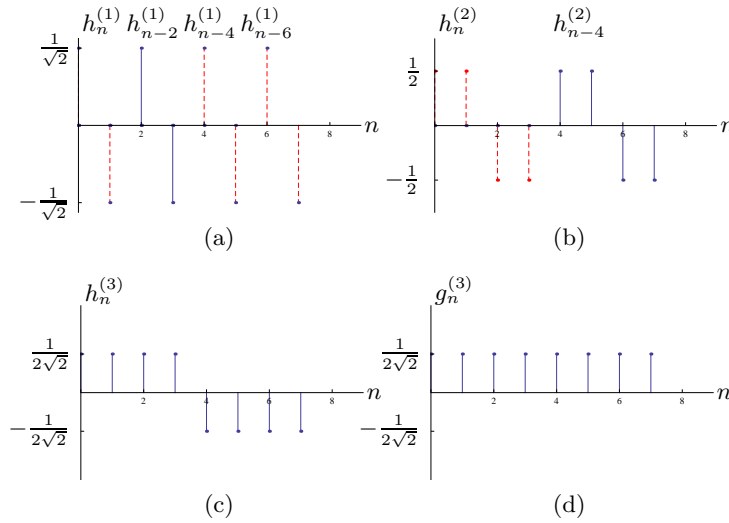
$$W^{(1)} = \overline{\text{span}}(\{h_{n-2k}^{(1)}\}_{k \in \mathbb{Z}}). \quad (9.3a)$$

The second channel, with input  $\beta^{(2)}$ , has the equivalent filter (9.1b) with upsampling by 4 in front. The corresponding sequences spanning the subspace  $W^{(2)}$  are (Figure 9.4(b)):

$$W^{(2)} = \overline{\text{span}}(\{h_{n-4k}^{(2)}\}_{k \in \mathbb{Z}}). \quad (9.3b)$$

The third and forth channels, with inputs  $\beta^{(3)}$  and  $\alpha^{(3)}$ , have the equivalent filters (9.1b), (9.1b), respectively, with upsampling by 8 in front. The corresponding sequences spanning the subspaces  $W^{(3)}$  and  $V^{(3)}$  are (Figure 9.4(c), (d)):

$$W^{(3)} = \overline{\text{span}}(\{h_{n-8k}^{(3)}\}_{k \in \mathbb{Z}}), \quad (9.3c)$$



**Figure 9.4:** Discrete-time Haar basis. Eight of the basis sequences forming  $\Phi_0$ : (a) level  $\ell = 1$ ,  $h_n^{(1)}$  and three of its shifts by 2, (b) level  $\ell = 2$ ,  $h_n^{(2)}$  and one of its shifts by 4, (c) level  $\ell = 3$ ,  $h_n^{(3)}$ , and (d) level  $\ell = 3$ ,  $g_n^{(3)}$ . A basis sequence at level  $i$  is orthogonal to a basis sequence at level  $j$ ,  $i < j$ , because it changes sign over an interval where the latter is constant (see, for example, the blue basis sequences).

$$V^{(3)} = \overline{\text{span}}(\{g_{n-8k}^{(3)}\}_{k \in \mathbb{Z}}). \quad (9.3d)$$

The complete set of basis sequences is thus:

$$\Phi = \{h_{n-2k}^{(1)}, h_{n-4k}^{(2)}, h_{n-8k}^{(3)}, g_{n-8k}^{(3)}\}_{k \in \mathbb{Z}}. \quad (9.3e)$$

**Orthogonality of Basis Sequences** While we have called the above sequence basis sequences, we have not established yet that they indeed form a basis (although this is almost obvious from the two-channel filter bank discussion).

The sets spanning  $W^{(1)}$ ,  $W^{(2)}$ ,  $W^{(3)}$  and  $V^{(3)}$  are all orthonormal sets, as the sequences within those sets do not overlap. To show that  $\Phi$  is an orthonormal set, we must show that sequences in each of the above subsets are orthogonal to each other. To prove that, we have to show that  $h^{(1)}$  and its shifts by 2 are orthogonal to  $h^{(2)}$  and its shifts by 4,  $h^{(3)}$  and its shifts by 8, and  $g^{(3)}$  and its shifts by 8. Similarly, we must show that  $h^{(2)}$  and its shifts by 4 are orthogonal to  $h^{(3)}$  and its shifts by 8 and  $g^{(3)}$  and its shifts by 8, etc. For Haar filters, this can be done by observing, for example, that  $h^{(1)}$  and its shifts by 2 always overlap a constant portion of  $h^{(2)}$ ,  $h^{(3)}$  and  $g^{(3)}$ , leading to a zero inner product (see Figure 9.4). With more general filters, this proof is more involved and will be considered later in the chapter.

To prove completeness, we first introduce the matrix view of this expansion.



**Matrix View** We have already seen that while  $g^{(3)}$  and  $h^{(3)}$  move in steps of 8,  $h^{(2)}$  moves in steps of 4 and  $h^{(1)}$  moves in steps of 2. That is, during the nonzero portion of  $g^{(3)}$  and  $h^{(3)}$ ,  $h^{(2)}$  and its shift by 4 occur, as well as  $h^{(1)}$  and its shifts by 2, 4 and 6 (see Figure 9.4). Thus, as in Chapter 7, we can describe the action of the filter bank via an infinite matrix:

$$\Phi = \text{diag}(\Phi_0), \quad (9.4)$$

with  $\Phi_0$  as

$$\Phi_0 = \begin{bmatrix} h_n^{(1)} & h_{n-2}^{(1)} & h_{n-4}^{(1)} & h_{n-6}^{(1)} & h_n^{(2)} & h_{n-4}^{(2)} & h_n^{(3)} & g_n^{(3)} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ -1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & -1 & 0 & 1 & 1 \\ 0 & -1 & 0 & 0 & -1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & -1 & 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & 0 & 1 & 0 & -1 & -1 & 1 \\ 0 & 0 & 0 & -1 & 0 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & & & & & & & \\ & \frac{1}{\sqrt{2}} & & & & & & \\ & & \frac{1}{\sqrt{2}} & & & & & \\ & & & \frac{1}{\sqrt{2}} & & & & \\ & & & & \frac{1}{2} & & & \\ & & & & & \frac{1}{2} & & \\ & & & & & & \frac{1}{2\sqrt{2}} & \\ & & & & & & & \frac{1}{2\sqrt{2}} \end{bmatrix}.$$

As before,  $\Phi$  is block diagonal only when the length of the filters in the original filter bank is equal to the downsampling factor, as is the case for Haar. The block is of length  $8 \times 8$  in this case, since the same structure repeats itself every 8 samples. That is,  $h^{(3)}$  and  $g^{(3)}$  repeat every 8 samples,  $h^{(2)}$  repeats every 4 samples, while  $h^{(1)}$  repeats every 2 samples. Thus, there will be 2 instances of  $h^{(2)}$  in block  $\Phi_0$  and 4 instances of  $h^{(1)}$  (see Figure 9.4). The basis sequences are the columns of the matrix  $\Phi$  at the center block  $\Phi_0$  and all their shifts by 8 (which corresponds to other blocks  $\Phi_0$  in  $\Phi$ ).  $\Phi$  is unitary, as each block  $\Phi_0$  is unitary, proving completeness for the Haar case. As we shall see, if each two-channel filter bank is orthonormal, even for longer filters, the orthonormality property will hold in general.

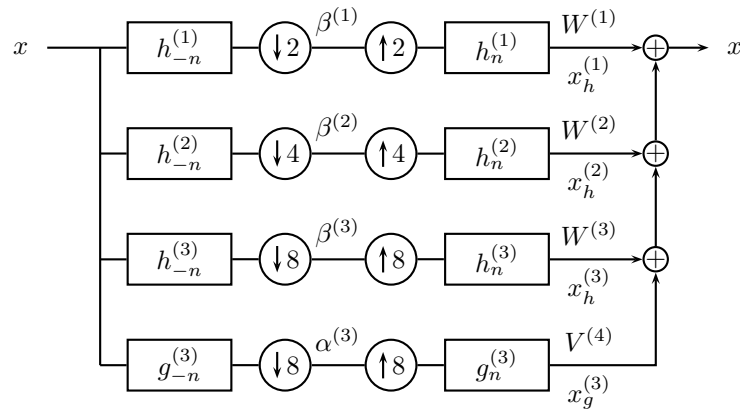
**Projection Properties** In summary, the 3-level iterated two-channel filter bank from Figure 9.1, splits the original space  $\ell^2(\mathbb{Z})$  into four subspaces:

$$\ell^2(\mathbb{Z}) = V^{(3)} \oplus W^{(3)} \oplus W^{(2)} \oplus W^{(1)},$$

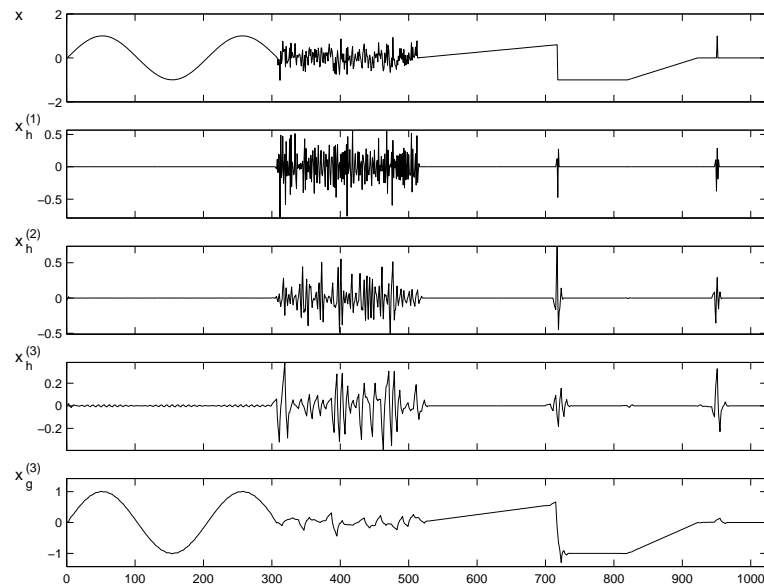
given in (9.3a)–(9.3d), again, a property that will hold in general. Figure 9.5 illustrates this split, where  $x_g^{(3)}$  denotes the projection onto  $V^{(3)}$ , and  $x_h^{(\ell)}$  denotes the projection onto  $W^{(\ell)}$ ,  $\ell = 1, 2, 3$ , while Figure 9.6 shows an example input sequence and the resulting channel sequences. The low-frequency sinusoid and the polynomial pieces are captured by the lowpass projection, white noise is apparent in all channels, and effects of discontinuities are localized in the bandpass channels.

## Chapter Outline

After this brief introduction, the structure of the chapter follows naturally. First, we generalize the Haar discussion and consider tree-structured filter banks, and,



**Figure 9.5:** Projection of the input sequence onto  $V^{(3)}$ ,  $W^{(3)}$ ,  $W^{(2)}$  and  $W^{(1)}$ , respectively, and perfect reconstruction as the sum of the projections.



**Figure 9.6:** Approximation properties of the discrete wavelet transform. (a) Original sequence  $x$  with various components. The highpass approximation after the (b) first iteration  $x_h^{(1)}$ , (c) second iteration  $x_h^{(2)}$ , and (d) the third iteration  $x_h^{(3)}$ . (e) The lowpass approximation  $x_g^{(3)}$ .

in particular, those that will lead to the DWT in Section 9.2. In Section 9.3, we study the orthogonal DWT and its properties such as approximation and projection capabilities. Section 9.4 discusses a variation on the DWT, the biorthogonal DWT, while Section 9.5 discusses another one, wavelet packets, which allow for rather arbitrary tilings of the time-frequency plane. We follow by computational aspects in Section 9.6.

## 9.2 Tree-Structured Filter Banks

Out of the basic building blocks of a two-channel filter bank (Chapter 7) and an  $N$ -channel filter bank (Chapter 8), we can build many different representations (Figure 9.7 shows some of the options together with the associated time-frequency tilings). We now set the stage by showing how to compute equivalent filters in such filter banks, as a necessary step towards building an orthogonal DWT in the next section, biorthogonal DWT in Section 9.4, wavelet packets in Section 9.5. We will assume we iterate orthogonal two-channel filter banks only (the analysis is parallel for biorthogonal and/or  $N$ -channel filter banks), and that  $J$  times. We consider the equivalent filter along the lowpass branches separately, followed by the bandpass branches, and finally, the relationship between the lowpass and bandpass ones. While we could make the discussion more general, we consider bandpass channels to be only those iterated through lowpass branches until the last step, when a final iteration is through a highpass one, as is the case for the DWT (iterations through arbitrary combinations of lowpass and highpass branches would follow similarly).

### 9.2.1 The Lowpass Channel and Its Properties

We start with the lowpass channel iterated  $J$  times, leading to  $g^{(J)}$ . Using the same identity to move the filter past the upsampler as in Figure 9.2, a cascade of  $J$  times upsampling and filtering by  $G(z)$  leads to upsampling by  $2^J$  followed by filtering with the equivalent filter

$$G^{(J)}(z) = G(z)G(z^2)G(z^4)\dots G(z^{2^{J-1}}) = \prod_{\ell=0}^{J-1} G(z^{2^\ell}), \quad (9.5a)$$

as shown in Figure 9.8. If  $g$  is of length  $L$ , then  $g^{(J)}$  is of length

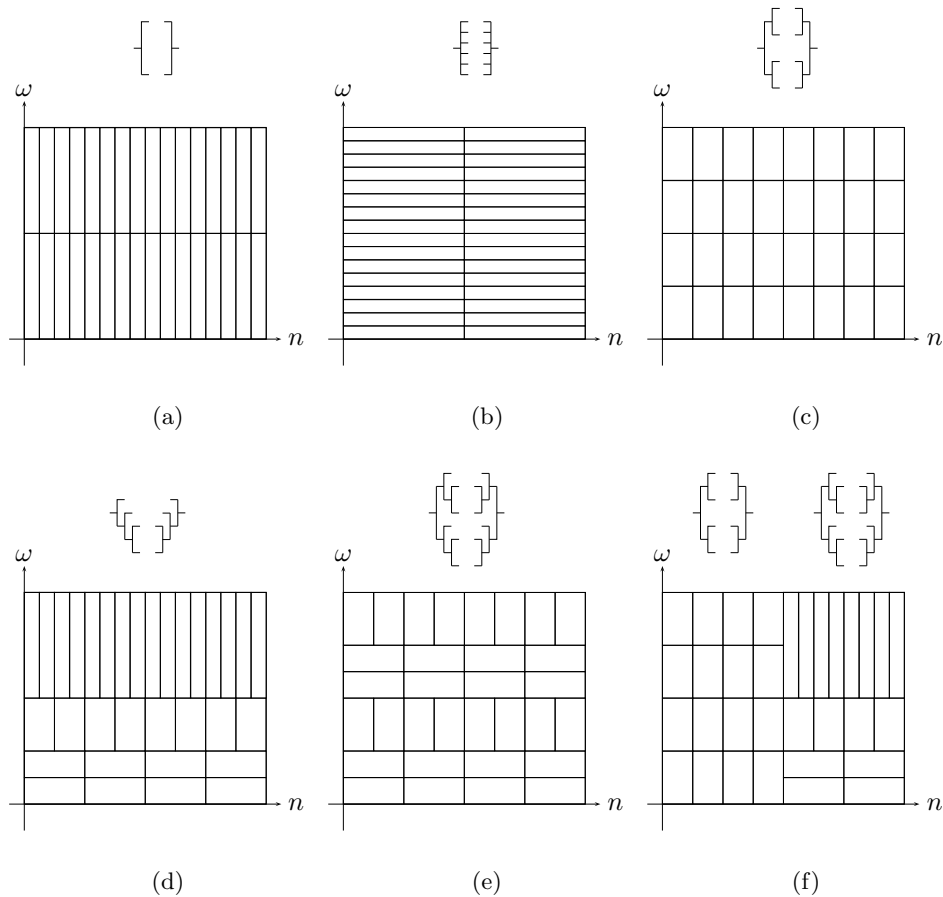
$$L^{(J)} = (L-1)(2^J - 1) + 1 \leq (L-1)2^J. \quad (9.5b)$$

Moreover, we see that

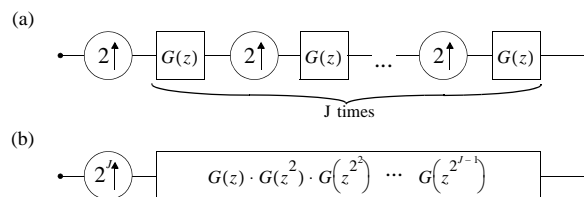
$$G^{(J)}(z) = G(z)G^{(J-1)}(z^2) = G^{(J-1)}(z)G(z^{2^{J-1}}). \quad (9.5c)$$

Some other recursive relations are given in Exercise 9.2.

**Orthogonality of the Lowpass Filter** We can use the orthogonality of the basic two-channel synthesis building block to show the orthogonality of the synthesis



**Figure 9.7:** Filter banks and variations together with the corresponding time-frequency tilings. (a) Two-channel filter bank (Chapter 7). (b)  $N$ -channel filter bank (Chapter 8). (c) The local Fourier transform filter bank (Chapter 8). (d) The DWT tree (present chapter). (e) The wavelet packet filter bank (present chapter). (f) The time-varying filter bank.



**Figure 9.8:** Cascade of  $J$  times upsampling and filtering. (a) Original system. (b) Equivalent system.

## 9.2. Tree-Structured Filter Banks

685

operator obtained by iterating. From this it must be true that the iterated lowpass filter is orthogonal to its translates by  $2^J$ . As in Section 7.2.1, we summarize the orthogonality properties here (as the proof is just a straightforward, albeit tedious, use of multirate operations, we leave it as Exercise 9.1):

$$\begin{array}{ccc}
 \langle g_n^{(J)}, g_{n-2^J k}^{(J)} \rangle = \delta_k & \begin{array}{c} \xleftrightarrow{\text{Matrix View}} \\ \xleftrightarrow{\text{ZT}} \\ \xleftrightarrow{\text{DTFT}} \end{array} & \begin{array}{l} D_{2^J}(G^{(J)})^T G^{(J)} U_{2^J} = I \\ \sum_{k=0}^{2^J-1} G^{(J)}(W_{2^J}^k z) G^{(J)}(W_{2^J}^{-k} z^{-1}) = 2^J \\ \sum_{k=0}^{2^J-1} |G^{(J)}(W_{2^J}^k e^{j\omega})|^2 = 2^J \end{array}
 \end{array} \quad (9.6a)$$

In the matrix view, we have used linear operators (infinite matrices) introduced in Section 2.7: (1) downsampling by  $2^J$ ,  $D_{2^J}$ , from (2.183); (2) upsampling by  $2^J$ ,  $U_{2^J}$ , from (2.188); and (3) filtering by  $G^{(J)}$ , from (2.63). The matrix view expresses the fact that the columns of  $G^{(J)} U_{2^J}$  form an orthonormal set. The DTFT version is a version of the quadrature mirror formula we have seen in (2.208). This filter is an  $2^J$ th-band filter (an ideal  $2^J$ th-band filter would be bandlimited to  $|\omega| \leq \pi/2^J$ , see Figure 9.1(c) with  $J = 3$ ).

Let us check the  $z$ -transform version of the above for  $J = 2$ :

$$\begin{aligned}
 & \sum_{k=0}^3 G^{(2)}(W_4^k z) G^{(2)}(W_4^{-k} z^{-1}) \\
 &= G^{(2)}(z) G^{(2)}(z^{-1}) + G^{(2)}(jz) G^{(2)}(-jz^{-1}) \\
 &\quad + G^{(2)}(-z) G^{(2)}(-z^{-1}) + G^{(2)}(-jz) G^{(2)}(jz^{-1}) \\
 &\stackrel{(a)}{=} G(z) G(z^2) G(z^{-1}) G(z^{-2}) + G(jz) G(-z^2) G(-jz^{-1}) G(-z^{-2}) \\
 &\quad + G(-z) G(z^2) G(-z^{-1}) G(z^{-2}) + G(-jz) G(-z^2) G(jz^{-1}) G(-z^{-2}) \\
 &\stackrel{(b)}{=} G(z^2) G(z^{-2}) \underbrace{(G(z) G(z^{-1}) + G(-z) G(-z^{-1}))}_2 \\
 &\quad + G(-z^2) G(-z^{-2}) \underbrace{(G(jz) G(-jz^{-1}) + G(-jz) G(jz^{-1}))}_2 \\
 &\stackrel{(c)}{=} 2 \underbrace{(G(z^2) G(z^{-2}) + G(-z^2) G(-z^{-2}))}_2 \stackrel{(d)}{=} 4,
 \end{aligned}$$

where (a) follows from the expression for the equivalent lowpass filter at level 2, (9.5a); in (b) we pulled out common terms  $G(z^2) G(z^{-2})$  and  $G(-z^2) G(-z^{-2})$ ; and (c) and (d) follow from the orthogonality of the lowpass filter  $g$ , (7.13). This, of course, is to be expected, because we have done nothing else but concatenate orthogonal filter banks, which we know already implement orthonormal bases, and thus, must satisfy orthogonality properties.

**Deterministic Autocorrelation of the Lowpass Filter** As we have done in Chapter 7, we rephrase the above results in terms of the deterministic autocorrelation of the filter. This is also what we use to prove (9.6a) in Exercise 9.1. The deterministic

---

**Lowpass Channel in a  $J$ -Level Octave-Band Orthogonal Filter Bank**


---

**Lowpass filter**

Original domain	$g_n^{(J)}$	$\langle g_n^{(J)}, g_{n-2^J k}^{(J)} \rangle_n = \delta_k$
Matrix domain	$G^{(J)}$	$D_{2^J} (G^{(J)})^T G^{(J)} U_{2^J} = I$
z-domain	$G^{(J)}(z) = \prod_{\ell=0}^{J-1} G(z^{2^\ell})$	$\sum_{k=0}^{2^J-1} G^{(J)}(W_{2^J}^k z) G^{(J)}(W_{2^J}^{-k} z^{-1}) = 2^J$
DTFT domain	$G^{(J)}(e^{j\omega})$	$\sum_{k=0}^{2^J-1}  G^{(J)}(W_{2^J}^k e^{j\omega}) ^2 = 2^J$

**Deterministic autocorrelation**

Original domain	$a_n^{(J)} = \langle g_k^{(J)}, g_{k+n}^{(J)} \rangle_k$	$a_{2^J k}^{(J)} = \delta_k$
Matrix domain	$A^{(J)} = (G^{(J)})^T G^{(J)}$	$D_{2^J} A^{(J)} U_{2^J} = I$
z-domain	$A^{(J)}(z) = G^{(J)}(z) G^{(J)}(z^{-1})$	$\sum_{k=0}^{2^J-1} A^{(J)}(W_{2^J}^k z) = 2^J$
DTFT domain	$A^{(J)}(e^{j\omega}) =  G^{(J)}(e^{j\omega}) ^2$	$\sum_{k=0}^{2^J-1} A^{(J)}(W_{2^J}^k e^{j\omega}) = 2^J$

<b>Orthogonal projection onto smooth space</b>	$V^{(J)} = \overline{\text{span}}(\{g_{n-2^J k}^{(J)}\}_{k \in \mathbb{Z}})$
$x_{V^{(J)}} = P_{V^{(J)}} x$	$P_V = G^{(J)} U_{2^J} D_{2^J} (G^{(J)})^T$

---

**Table 9.1:** Properties of the lowpass channel in an orthogonal  $J$ -level octave-band filter bank.

autocorrelation of  $g^{(J)}$  is denoted by  $a^{(J)}$ .

$$\begin{array}{ccc}
 \langle g_n^{(J)}, g_{n-2^J k}^{(J)} \rangle & = & a_{2^J k}^{(J)} = \delta_k \\
 \begin{array}{c} \text{Matrix View} \\ \longleftrightarrow \\ \text{ZT} \\ \longleftrightarrow \\ \text{DTFT} \end{array} & & \begin{array}{l} D_{2^J} A^{(J)} U_{2^J} = I \\ \sum_{k=0}^{2^J-1} A^{(J)}(W_{2^J}^k z) = 2^J \\ \sum_{k=0}^{2^J-1} A^{(J)}(W_{2^J}^k e^{j\omega}) = 2^J \end{array}
 \end{array} \quad (9.6b)$$

**Orthogonal Projection Property of the Lowpass Channel** We now look at the lowpass channel as a composition of four linear operators we just saw:

$$x_{V^{(J)}} = P_{V^{(J)}} x = G^{(J)} U_{2^J} D_{2^J} (G^{(J)})^T x. \quad (9.7)$$

As before, the notation is evocative of projection onto  $V^{(J)}$ , and we will now show that the lowpass channel accomplishes precisely this. Using (9.6a), we check idem-

potency and self-adjointness of  $P_{V^{(J)}}$  (Definition 1.27),

$$\begin{aligned} P_{V^{(J)}}^2 &= (G^{(J)} U_{2^J} \overbrace{D_{2^J} (G^{(J)})^T}^I (G^{(J)} U_{2^J} D_{2^J} (G^{(J)})^T) \\ &\stackrel{(a)}{=} G^{(J)} U_{2^J} D_{2^J} (G^{(J)})^T = P_{V^{(J)}}, \\ P_{V^{(J)}}^T &= (G^{(J)} U_{2^J} D_{2^J} (G^{(J)})^T)^T = G^{(J)} (U_{2^J} D_{2^J})^T (G^{(J)})^T \\ &\stackrel{(b)}{=} G^{(J)} U_{2^J} D_{2^J} (G^{(J)})^T = P_{V^{(J)}}, \end{aligned}$$

where (a) follows from (9.6a) and (b) from (2.190). Indeed,  $P_{V^{(J)}}$  is an orthogonal projection operator, with the range given in:

$$V^{(J)} = \overline{\text{span}}(\{g_{n-2^J k}^{(J)}\}_{k \in \mathbb{Z}}). \quad (9.8)$$

The summary of properties of the lowpass channel is given in Table 9.1.

### 9.2.2 Bandpass Channels and Their Properties

While we have only one lowpass filter, in a  $J$ -level octave-band filter bank leading to the DWT, we also have  $J$  bandpass filters, ideally, each bandlimited to  $\pi/2^{\ell+1} \leq |\omega| \leq \pi/2^\ell$ , for  $\ell = 0, 1, \dots, J-1$ , as in Figure 9.1(c) with  $J = 3$ . The analysis of an iterated filter bank constructed through arbitrary combinations of lowpass and highpass branches would follow similarly.

The filter  $H^{(\ell)}(z)$  corresponds to a branch with a highpass filter followed by  $(\ell - 1)$  lowpass filters (always with upsampling by 2 in between). The  $(\ell - 1)$ th lowpass filter branch has an equivalent filter  $G^{(\ell-1)}(z)$  as in (9.5a), preceded by upsampling by  $2^{\ell-1}$ . Passing this upsampling across the initial highpass filter changes  $H(z)$  into  $H(z^{2^{\ell-1}})$  and

$$H^{(\ell)}(z) = H(z^{2^{\ell-1}}) G^{(\ell-1)}(z), \quad i = 1, \dots, J, \quad (9.9)$$

follows. The basis vectors correspond to the impulse responses and the shifts given by the upsampling factors. These upsampling factors are  $2^J$  for the lowpass branch and  $2^\ell$  for the bandpass branches.

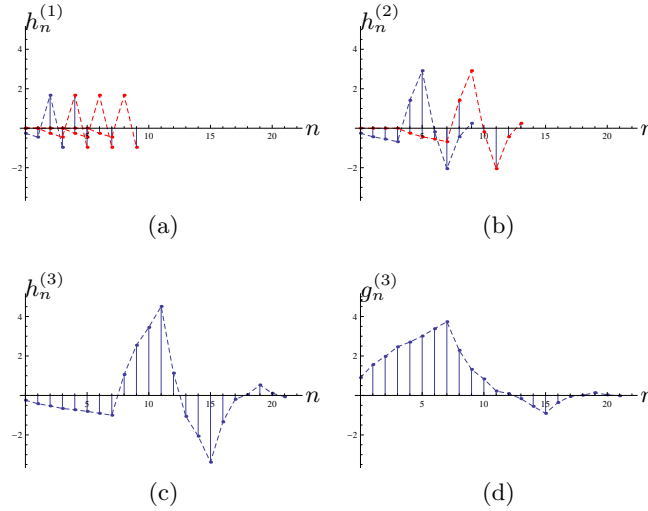
**EXAMPLE 9.1 (THE 3-LEVEL OCTAVE-BAND DAUBECHIES FILTER BANK)** We continue Example 7.3, the Daubechies filter with two zeros at  $z = -1$ :

$$G(z) = \frac{1}{4\sqrt{2}} \left[ (1 + \sqrt{3}) + (3 + \sqrt{3})z^{-1} + (3 - \sqrt{3})z^{-2} + (1 - \sqrt{3})z^{-3} \right]. \quad (9.10)$$

A 3-level octave-band filter bank has as basis sequences the impulse responses of:

$$\begin{aligned} H^{(1)}(z) &= H(z) = -z^{-3}G(-z^{-1}), \\ H^{(2)}(z) &= G(z)H(z^2), \\ H^{(3)}(z) &= G(z)G(z^2)H(z^4), \\ G^{(3)}(z) &= G(z)G(z^2)G(z^4), \end{aligned}$$

together with shifts by multiples of 2, 4, 8 and 8, respectively (see Figure 9.9).



**Figure 9.9:** Basis sequences for a 3-level octave-band filter bank based on a Daubechies orthonormal length-4 filter with two zeros at  $z = -1$ , as in (9.10). The basis sequences are  $h^{(1)}, h^{(2)}, h^{(3)}$  and  $g^{(3)}$ , together with shifts by multiples of 2, 4, 8 and 8, respectively. We show (a)  $h_n^{(1)}, h_{n-2}^{(1)}, h_{n-4}^{(1)}, h_{n-6}^{(1)}$ , (b)  $h_n^{(2)}, h_{n-4}^{(2)}$ , (c)  $h_n^{(3)}$ , and (d)  $g_n^{(3)}$ .

**Orthogonality of an Individual Bandpass Filter** Unlike in the simple two-channel case, now, each bandpass filter  $h^{(\ell)}$  is orthogonal to its shifts by  $2^\ell$ , and to all the other bandpass filters  $h^{(j)}$ ,  $\ell \neq j$ , as well as to their shifts by  $2^j$ . We expect these to hold as they hold in the basic building block, but state them nevertheless. While we could state together the orthogonality properties for a single level and across levels, we separate them for clarity. All the proofs are left for Exercise 9.1.

$$\begin{aligned}
 \langle h_n^{(\ell)}, h_{n-2^\ell k}^{(\ell)} \rangle &= \delta_k & \begin{array}{l} \text{Matrix View} \\ \longleftrightarrow \\ \text{ZT} \\ \longleftrightarrow \\ \text{DTFT} \\ \longleftrightarrow \end{array} & \begin{aligned} D_{2^\ell} (H^{(\ell)})^T H^{(\ell)} U_{2^\ell} &= I \\ \sum_{k=0}^{2^\ell-1} H^{(\ell)}(W_{2^\ell}^k z) H^{(\ell)}(W_{2^\ell}^{-k} z^{-1}) &= 2^\ell \\ \sum_{k=0}^{2^\ell-1} |H^{(\ell)}(W_{2^\ell}^k e^{j\omega})|^2 &= 2^\ell \end{aligned} \end{aligned} \quad (9.11a)$$

**Orthogonality of Different Bandpass Filters** Without loss of generality, let us assume that  $\ell < j$ . We summarize the orthogonality properties of the bandpass filter  $h^{(\ell)}$  and its shift by  $2^\ell$  to the bandpass filter  $h^{(j)}$  and its shift by  $2^j$ .

$$\begin{aligned}
 \langle h_{n-2^\ell k}^{(\ell)}, h_{n-2^j k}^{(j)} \rangle &= 0 & \begin{array}{l} \text{Matrix View} \\ \longleftrightarrow \\ \text{ZT} \\ \longleftrightarrow \\ \text{DTFT} \\ \longleftrightarrow \end{array} & \begin{aligned} D_{2^j} (H^{(j)})^T H^{(\ell)} U_{2^\ell} &= 0 \\ \sum_{k=0}^{2^\ell-1} H^{(\ell)}(W_{2^\ell}^k z) H^{(j)}(W_{2^\ell}^{-k} z^{-1}) &= 0 \\ \sum_{k=0}^{2^\ell-1} H^{(\ell)}(W_{2^\ell}^k e^{j\omega}) H^{(j)}(W_{2^\ell}^k e^{-j\omega}) &= 0 \end{aligned} \end{aligned} \quad (9.11b)$$



**Deterministic Autocorrelation of Individual Bandpass Filters**

$$\begin{array}{ccc}
\langle h_n^{(\ell)}, h_{n-2^\ell k}^{(\ell)} \rangle = a_{2^\ell k}^{(\ell)} = \delta_k & \begin{array}{c} \text{Matrix View} \\ \longleftrightarrow \\ \text{ZT} \\ \longleftrightarrow \\ \text{DTFT} \end{array} & \begin{array}{l} D_{2^\ell} A^{(\ell)} U_{2^\ell} = I \\ \sum_{k=0}^{2^\ell-1} A^{(\ell)}(W_{2^\ell}^k z) = 2^\ell \\ \sum_{k=0}^{2^\ell-1} A^{(\ell)}(W_{2^\ell}^k e^{j\omega}) = 2^\ell \end{array} \quad (9.11c)
\end{array}$$

**Deterministic Crosscorrelation of Different Bandpass Filters** Again without loss of generality, we assume that  $\ell < j$ .

$$\begin{array}{ccc}
\langle h_{n-2^\ell k}^{(\ell)}, h_{n-2^j k}^{(j)} \rangle = c_{2^\ell k}^{(\ell,j)} = 0 & \begin{array}{c} \text{Matrix View} \\ \longleftrightarrow \\ \text{ZT} \\ \longleftrightarrow \\ \text{DTFT} \end{array} & \begin{array}{l} D_{2^j} C^{(\ell,j)} U_{2^\ell} = 0 \\ \sum_{k=0}^{2^\ell-1} C^{(\ell,j)}(W_{2^\ell}^k z) = 0 \\ \sum_{k=0}^{2^\ell-1} C^{(\ell,j)}(W_{2^\ell}^k e^{j\omega}) = 0 \end{array} \quad (9.11d)
\end{array}$$

**Orthogonal Projection Property of Bandpass Channels** The lowpass channel computes a projection onto a space of coarse sequences spanned by  $g^{(J)}$  and its shifts by  $2^J$ . Similarly, each bandpass channel computes a projection onto a space of detail sequences spanned by each of  $h^{(\ell)}$  and its shifts by  $2^\ell$ , for  $\ell = 1, 2, \dots, J$ . That is, we have  $J$  bandpass projection operators, computing bandpass projections:

$$x_{W^\ell} = P_{W^{(\ell)}} x = H^{(\ell)} U_{2^\ell} D_{2^\ell} (H^{(\ell)})^T x, \quad (9.12)$$

for  $\ell = 1, 2, \dots, J$ . That  $P_{W^{(\ell)}}$  is an orthogonal projection operator is easy to show; follow the same path as for the lowpass filter. Each bandpass space is given by:

$$W^{(\ell)} = \overline{\text{span}}(\{h_{n-2^\ell k}^{(\ell)}\}_{k \in \mathbb{Z}}). \quad (9.13)$$

**9.2.3 Relationship between Lowpass and Bandpass Channels**

The only conditions left to show for the lowpass impulse response and its shifts by  $2^J$  and all the bandpass impulse responses and their appropriate shifts to form an orthonormal set, is the orthogonality of the lowpass and bandpass sequences. Since the proofs follow the same path as before, we again leave them for Exercise 9.1.

**Orthogonality of the Lowpass and Bandpass Filters**

$$\begin{array}{ccc}
\langle g_{n-2^J k}^{(J)}, h_{n-2^\ell k}^{(\ell)} \rangle = 0 & \begin{array}{c} \text{Matrix View} \\ \longleftrightarrow \\ \text{ZT} \\ \longleftrightarrow \\ \text{DTFT} \end{array} & \begin{array}{l} D_{2^\ell} (H^{(\ell)})^T G^{(J)} U_{2^J} = 0 \\ \sum_{k=0}^{2^\ell-1} G^{(J)}(W_{2^\ell}^k z) H^{(\ell)}(W_{2^\ell}^{-k} z^{-1}) = 0 \\ \sum_{k=0}^{2^\ell-1} G^{(J)}(W_{2^\ell}^k e^{j\omega}) H^{(\ell)}(W_{2^\ell}^k e^{-j\omega}) = 0 \end{array} \quad (9.14a)
\end{array}$$

**Deterministic Crosscorrelation of the Lowpass and Bandpass Filters**

$$\begin{array}{ccc}
\langle g_{n-2^J k}^{(J)}, h_{n-2^\ell k}^{(\ell)} \rangle = c_{2^\ell k}^{(J,\ell)} = 0 & \begin{array}{c} \text{Matrix View} \\ \longleftrightarrow \\ \text{ZT} \\ \longleftrightarrow \\ \text{DTFT} \end{array} & \begin{array}{l} D_{2^\ell} C^{(J,\ell)} U_{2^J} = 0 \\ \sum_{k=0}^{2^\ell-1} C^{(J,\ell)}(W_{2^\ell}^k z) = 0 \\ \sum_{k=0}^{2^\ell-1} C^{(J,\ell)}(W_{2^\ell}^k e^{j\omega}) = 0 \end{array} \quad (9.14b)
\end{array}$$

### 9.3 Orthogonal Discrete Wavelet Transform

Following our introductory Haar example, it is now quite clear what the DWT does: it produces a coarse projection coefficient  $\alpha^{(J)}$ , together with a sequence of ever finer detail projection (wavelet) coefficients  $\beta^{(1)}, \beta^{(2)}, \dots, \beta^{(J)}$ , using a  $J$ -level octave-band filter bank as a vehicle. As we have seen in that simple example, the original space is split into a sequence of subspaces, each having a spectrum half the size of the previous (octave-band decomposition). Such a decomposition is appropriate for smooth sequences with isolated discontinuities (natural images are one example of such signals; there is evidence that the human visual system processes visual information in such a manner exactly).

#### 9.3.1 Definition of the Orthogonal DWT

We are now ready to formally define the orthogonal DWT:

**DEFINITION 9.1 (ORTHOGONAL DWT)** The  $J$ -level orthogonal DWT of a sequence  $x$  is a function of  $\ell \in \{1, 2, \dots, J\}$  given by

$$\alpha_k^{(J)} = \langle x_n, g_{n-2^J k}^{(J)} \rangle_n = \sum_{n \in \mathbb{Z}} x_n g_{n-2^J k}^{(J)}, \quad k \in \mathbb{Z}, \quad (9.15a)$$

$$\beta_k^{(\ell)} = \langle x_n, h_{n-2^\ell k}^{(\ell)} \rangle_n = \sum_{n \in \mathbb{Z}} x_n h_{n-2^\ell k}^{(\ell)}, \quad \ell \in \{1, 2, \dots, J\}. \quad (9.15b)$$

The inverse DWT is given by

$$x_n = \sum_{k \in \mathbb{Z}} \alpha_k^{(J)} g_{n-2^J k}^{(J)} + \sum_{\ell=1}^J \sum_{k \in \mathbb{Z}} \beta_k^{(\ell)} h_{n-2^\ell k}^{(\ell)}. \quad (9.15c)$$

In the above, the  $\alpha^{(J)}$  are the *scaling coefficients* and the  $\beta^{(\ell)}$  are the *wavelet coefficients*.

The equivalent filter  $g^{(J)}$  is often called the *scaling sequence* and  $h^{(\ell)}$  *wavelets* (wavelet sequences),  $\ell = 1, 2, \dots, J$ ; they are given in (9.5a) and (9.9), respectively, and satisfy (9.6a)–(9.6b), (9.11a)–(9.11c), as well as (9.14a)–(9.14b).

To denote such a DWT pair, we write:

$$x_n \xleftrightarrow{\text{DWT}} \alpha_k^{(J)}, \beta_k^{(J)}, \beta_k^{(J-1)}, \dots, \beta_k^{(1)}.$$

The orthogonal DWT is implemented using a  $J$ -level octave-band orthogonal filter bank as in Figure 9.1. This particular version of the DWT is called the *dyadic* DWT as each subsequent channel has half of the coefficients of the previous one. Various generalizations are possible; for example, Solved Exercise 9.2 considers the DWT obtained from a 3-channel filter bank.

### 9.3.2 Properties of the Orthogonal DWT

Some properties of the DWT are rather obvious (such as linearity), while others are more involved (such as shift in time). We now list and study a few of these.

DWT properties	Time domain	DWT domain
Linearity	$ax_n + by_n$	$a \{\alpha_{x,k}^{(J)}, \beta_{x,k}^{(J)}, \dots, \beta_{x,k}^{(1)}\} + b \{\alpha_{y,k}^{(J)}, \beta_{y,k}^{(J)}, \dots, \beta_{y,k}^{(1)}\}$
Shift in time	$x_{n-2^J n_0}$	$\alpha_{k-n_0}^{(J)}, \beta_{k-n_0}^{(J)}, \beta_{k-2n_0}^{(J-1)}, \dots, \beta_{k-2^{J-1}n_0}^{(1)}$
Parseval's equality	$\ x\ ^2 = \sum_{n \in \mathbb{Z}}  x_n ^2$	$= \ \alpha^{(J)}\ ^2 + \sum_{\ell=1}^J \ \beta^{(\ell)}\ ^2$

**Table 9.2:** Properties of the DWT.

**Linearity** The DWT operator is a linear operator, or,

$$ax_n + by_n \xleftrightarrow{\text{DWT}} a \{\alpha_{x,k}^{(J)}, \beta_{x,k}^{(J)}, \dots, \beta_{x,k}^{(1)}\} + b \{\alpha_{y,k}^{(J)}, \beta_{y,k}^{(J)}, \dots, \beta_{y,k}^{(1)}\}. \quad (9.16)$$

**Shift in Time** A shift in time by  $2^J n_0$  results in

$$x_{n-2^J n_0} \xleftrightarrow{\text{DWT}} \alpha_{k-n_0}^{(J)}, \beta_{k-n_0}^{(J)}, \beta_{k-2n_0}^{(J-1)}, \dots, \beta_{k-2^{J-1}n_0}^{(1)}. \quad (9.17)$$

This property shows that the DWT is not shift invariant; it is periodically shift variant with the period  $2^J$ .

**Parseval's Equality** The DWT operator is a unitary operator and thus preserves the Euclidean norm (see (1.51)):

$$\|x\|^2 = \sum_{n \in \mathbb{Z}} |x_n|^2 = \|\alpha^{(J)}\|^2 + \sum_{\ell=1}^J \|\beta^{(\ell)}\|^2. \quad (9.18)$$

**Projection** After our Haar example, it should come as no surprise that a  $J$ -level orthogonal DWT projects the input sequence  $x$  onto one lowpass space

$$V^{(J)} = \overline{\text{span}}(\{g_{n-2^J k}^{(J)}\}_{k \in \mathbb{Z}}),$$

and  $J$  bandpass spaces

$$W^{(\ell)} = \overline{\text{span}}(\{h_{n-2^\ell k}^{(\ell)}\}_{k \in \mathbb{Z}}), \quad \ell = 1, \dots, J,$$

where  $g^{(J)}$  and  $h^{(\ell)}$  are the equivalent filters given in (9.5a) and (9.9), respectively. The input space  $\ell^2(\mathbb{Z})$  is split into the following  $(J+1)$  spaces:

$$\ell^2(\mathbb{Z}) = V^{(J)} \oplus W^{(J)} \oplus W^{(J-1)} \oplus \dots \oplus W^{(1)}.$$

**Polynomial Approximation** As we have done in Section 7.2.5, we now look at polynomial approximation properties of an orthogonal DWT,<sup>127</sup> under the assumption that the lowpass filter  $g$  has  $N \geq 1$  zeros at  $z = -1$ , as in (7.46):

$$G(z) = (1 + z^{-1})^N R(z).$$

Note that  $R(z)|_{z=-1}$  cannot be zero because of the orthogonality constraint (7.13). Remember that the highpass filter, being a modulated version of the lowpass, has  $N$  zeros at  $z = 1$ . In other words, it annihilates polynomials up to degree  $(N - 1)$  since it takes an  $N$ th-order difference of the sequence.

In the DWT, each bandpass channel annihilates finitely-supported polynomials of a certain degree, which are therefore carried by the lowpass branch. That is, if  $x$  is a polynomial sequence of degree smaller than  $N$ , the channel sequences  $\beta^{(i)}$  are all zero, and that polynomial sequence  $x$  is projected onto  $V^{(J)}$ , the lowpass approximation space:

$$x_n = n^i = \sum_{k \in \mathbb{Z}} \alpha_k^{(J)} g_{n-2^J k}^J, \quad 0 < i < N,$$

that is, the equivalent lowpass filter reproduces polynomials up to degree  $(N - 1)$ . As this is an orthogonal DWT, the scaling coefficients follow from (9.15a),

$$\alpha_k^{(J)} = \langle x_n, g_{n-2^J k}^{(J)} \rangle_n = \langle n^i, g_{n-2^J k}^{(J)} \rangle_n = \sum_{n \in \mathbb{Z}} n^i g_{n-2^J k}^{(J)}.$$

An example with the 4-tap Daubechies orthogonal filter from (9.10) is given in Figure 9.10. Part (a) shows the equivalent filter after 6 levels of iteration:  $J = 6$ ,  $G^{(6)}(z) = G(z)G(z^2)G(z^4)G(z^8)G(z^{16})G(z^{32})$  and length  $L^{(6)} = 190$  from (9.5b). Part (b) shows the reproduction of a linear polynomial (over a finite range, ignoring boundary effects).

In summary, the DWT, when wavelet basis sequences have zero moments, will have very small inner products with smooth parts of an input sequence (and exactly zero when the sequence is locally polynomial). This will be one key ingredient in building successful approximation schemes using the DWT in Chapter 13.

**Characterization of Singularities** Due to its localization properties, the DWT has a unique ability to characterize singularities.

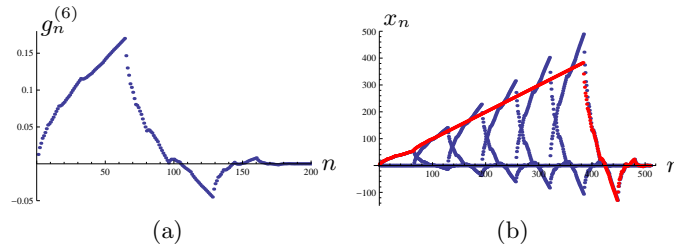
Consider a single nonzero sample in the input,  $x_n = \delta_{n-k}$ . This delayed Kronecker delta sequence now excites each equivalent filter's impulse response of length  $L^{(i)} = (L - 1)(2^i - 1) + 1$  at level  $\ell$  (see (9.5b)), which is then downsampled by  $2^i$  (see Figure 9.5 for an illustration with  $J = 3$ ). Thus, this single nonzero input creates at most  $(L - 1)$  nonzero coefficients in each channel. Furthermore, since each equivalent filter is of norm 1 and downsampled by  $2^\ell$ , the energy resulting at level  $\ell$  is of the order

$$\|\beta^{(\ell)}\|^2 \sim 2^{-\ell}.$$

<sup>127</sup>Recall that we are dealing with finitely-supported polynomial sequences, ignoring the boundary issues. If this were not the case, these sequences would not belong to any  $\ell^p$  space.

## 9.3. Orthogonal Discrete Wavelet Transform

693



**Figure 9.10:** Polynomial reproduction, exact over a finite range. (a) Equivalent filter's impulse response after six iterations. (b) Reproduction of a linear polynomial (in red) over a finite range. We also show the underlying weighted basis sequences  $(\alpha_k^{(6)} g_{n-2^j k}^{(6)}, k = 0, 1, \dots, 5)$  contributing to the reproduction of the polynomial. While the plots are all discrete, they give the impression of being connected due to point density.

In other words, the energy of the Kronecker delta sequence is roughly spread across the channels according to a geometric distribution. Another way to phrase the above result is to note that as  $\ell$  increases, coefficients  $\beta^{(\ell)}$  decay roughly as

$$\beta^{(\ell)} \sim 2^{-\ell/2},$$

when the input is an isolated Kronecker delta.

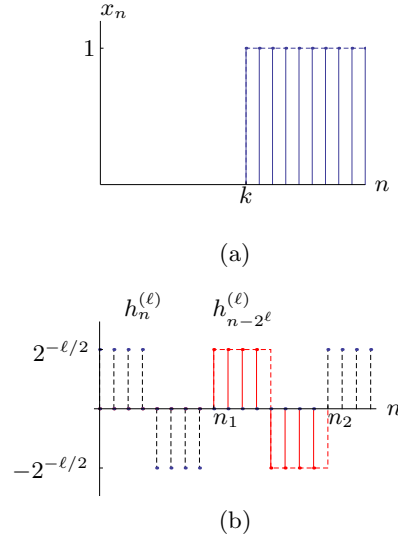
For a piecewise constant sequence, the coefficients behave instead as

$$\beta^{(\ell)} \sim 2^{\ell/2}.$$

Thus, two different types of singularities lead to different behaviors of wavelet coefficients across scales. In other words, if we can observe the behavior of wavelet coefficients across scales, we can make an educated guess of the type of singularity present in the input sequence, as we illustrate in Example 9.2. We will study this in more in detail in the continuous-time case, in Chapter 12.

**EXAMPLE 9.2 (CHARACTERIZATION OF SINGULARITIES BY THE HAAR DWT)** Convolution of the Kronecker delta sequence at position  $k$  with the Haar analysis filter  $h_{-n}^{(\ell)}$  in (9.2c) generates  $2^\ell$  coefficients; downsampling by  $2^\ell$  then leaves a single coefficient of size  $2^{-\ell/2}$ .

As an example of a piecewise constant sequence, we use the Heaviside sequence (2.10) delayed by  $k$ . A single wavelet coefficient will be different from zero at each scale, the one corresponding to the wavelet that straddles the discontinuity (Figure 9.11). At scale  $2^\ell$ , this corresponds to the wavelet with support from  $2^\ell \lfloor k/2^\ell \rfloor$  to  $2^\ell (\lfloor k/2^\ell \rfloor + 1)$ . All other wavelet coefficients are zero; on the left of the discontinuity because the sequence is zero, and on the right because the inner product is zero. The magnitude of the nonzero coefficient depends on the location  $k$  and varies between 0 and  $2^{\ell/2-1}$ . When  $k$  is a multiple of  $2^\ell$ , this magnitude is zero, and when  $k$  is equal to  $\ell 2^\ell + 2^{\ell/2}$ , it achieves its maximum value. The latter occurs when the discontinuity is aligned with the discontinuity



**Figure 9.11:** Characterization of singularities by the Haar DWT. (a) A Heaviside sequence at location  $k$ , and (b) the equivalent wavelet sequences, highpass filter  $h^{(\ell)}$  and its shifts, at scale  $2^\ell$ . A single wavelet, with support from  $n_1 = 2^\ell \lfloor k/2^\ell \rfloor$  to  $n_2 = 2^\ell (\lfloor k/2^\ell \rfloor + 1)$ , has a nonzero inner product with a magnitude of the order of  $2^{\ell/2}$ .

of the wavelet itself; then, the inner product is  $2^{\ell-1}2^{-\ell/2} = 2^{(\ell/2)-1}$ , and we obtain

$$\beta^{(\ell)} \sim 2^{(\ell/2)-1}.$$

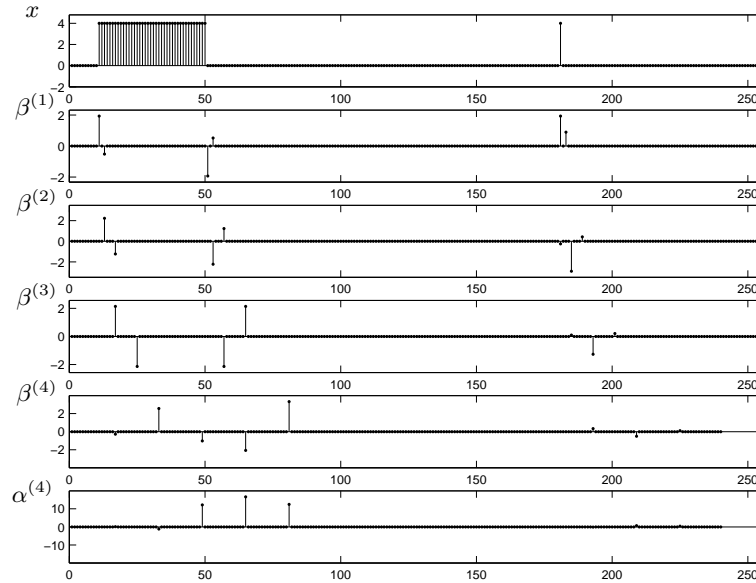
We thus see that the magnitudes of the wavelet coefficients will vary, but as  $\ell$  increases, they will increase at most as  $2^{\ell/2}$ . In Figure 9.12(a), we show an example input sequence consisting of a piecewise constant sequence and a Kronecker delta sequences, and its DWT in (b). We see that the wavelet coefficients are gathered around the singular points (Kronecker delta, Heaviside step), and they decay or increase, depending on the type of singularity.

In the example above, we obtained precisely  $\beta^{(\ell)} = 2^{(\ell/2)-1}$  for the one nonzero wavelet coefficient at scale  $2^\ell$ . Figure 9.12 gives another example, with a sequence with more types of singularities and a DWT with longer filters. We again have  $\sim 2^{-\ell/2}$  scaling of wavelet coefficient magnitudes and a roughly constant number of nonzero wavelet coefficients per scale. We will study this effect in more detail in Chapter 13, where the bounds on the coefficient magnitudes will play a large role in quantifying approximation performance.

In summary, the DWT acts as a *singularity detector*, that is, it leads to nonzero wavelet coefficients around singular points of a sequence. The number of nonzero coefficients per scale is bounded by  $(L-1)$ . Moreover, the magnitude of the wavelet coefficients across scales is an indicator of the type of singularity. Together with its

## 9.3. Orthogonal Discrete Wavelet Transform

695



**Figure 9.12:** A piecewise constant sequence plus a Kronecker delta sequence and its DWT. (a) The original sequence  $x$ . (b)–(e) Wavelet coefficients  $\beta^{(\ell)}$  at scales  $2^\ell$ ,  $\ell = 1, 2, 3, 4$ . (f) Scaling coefficients  $\alpha^{(4)}$ . To compare different channels, all the sequences have been upsampled by a factor  $2^\ell$ ,  $\ell = 1, 2, 3, 4$ .

polynomial approximation properties as described previously, this ability to characterize singularities will be the other key ingredient in building successful approximation schemes using the DWT in Chapter 13.

**Basis for  $\ell^2(\mathbb{Z})$**  An interesting twist on a  $J$ -level DWT is what happens when we let the number of levels  $J$  go to infinity. This will be one way we will be building continuous-time wavelet bases in Chapter 12. For sequences in  $\ell^2(\mathbb{Z})$ , such an infinitely-iterated DWT can actually build an orthonormal basis based on wavelet sequences (equivalent highpass filters) alone. The energy of the scaling coefficients vanishes in  $\ell^2$  norm; in other words, the original sequence is entirely captured by the wavelet coefficients, thus proving Parseval's equality for such a basis. While this is true in general, below we prove the result for Haar filters only; the general proof needs additional technical conditions that are beyond the scope of our text.

**THEOREM 9.2 (DISCRETE HAAR WAVELETS AS A BASIS FOR  $\ell^2(\mathbb{Z})$ )** The discrete-time wavelets  $h^{(\ell)}$  with impulse responses as in (9.2d) and their shifts by  $2^\ell$ ,

$$\Phi = \{h_{n-2^\ell k}^{(\ell)}\}_{k \in \mathbb{Z}, \ell=1, 2, \dots},$$

form an orthonormal basis for the space of finite-energy sequences,  $\ell^2(\mathbb{Z})$ .

*Proof.* To prove  $\Phi$  is an orthonormal basis, we must prove it is an orthonormal set and that it is complete. The orthonormality of basis functions was shown earlier in Section 9.1 for Haar filters and in Section 9.2 for the more general ones.

To prove completeness, we will show that Parseval's equality holds, that is, for an arbitrary input  $x \in \ell^2(\mathbb{Z})$ , we have

$$\|x\|^2 = \sum_{\ell=1}^{\infty} \|\beta^{(\ell)}\|^2, \quad (9.19)$$

where  $\beta^{(\ell)}$  are the wavelet coefficients at scales  $2^\ell$ ,  $\ell = 1, 2, \dots$ :

$$\beta_k^{(\ell)} = \langle x_n, h_{n-2^\ell k}^{(\ell)} \rangle_n.$$

For any finite number of decomposition levels  $J$ , the Parseval's equality (9.18) holds. Thus, our task is to show  $\lim_{J \rightarrow \infty} \|\alpha^{(J)}\|^2 = 0$ . We show this by bounding two quantities: the energy lost in truncating  $x$  and the energy in the scaling coefficients that represent the truncated sequence.

Without loss of generality, assume  $x$  has unit norm. For any  $\epsilon > 0$ , we will show that  $\|\alpha^{(J)}\|^2 < \epsilon$  for sufficiently large  $J$ . First note that there exists a  $K$  such that the restriction of  $x$  to  $\{-2^K, -2^K + 1, \dots, 0, 1, \dots, 2^K - 1\}$  has energy at least  $1 - \epsilon/2$ ; this follows from the convergence of the series defining the  $\ell^2$  norm of  $x$ . Denote the restriction by  $\tilde{x}$ .

A  $K$ -level decomposition of  $\tilde{x}$  has at most two nonzero scaling coefficients  $\tilde{\alpha}_{-1}^{(K)}$  and  $\tilde{\alpha}_0^{(K)}$ . Each of these scaling coefficients satisfies  $|\tilde{\alpha}_k^{(K)}| \leq 1$  because  $\|\tilde{\alpha}^{(K)}\|^2 \leq \|\tilde{x}\|^2 = 1$  by Bessel's inequality. We will now consider further levels of decomposition beyond  $K$ . After one more level, the lowpass output  $\tilde{\alpha}^{(K+1)}$  has coefficients

$$\tilde{\alpha}_0^{(K+1)} = \frac{1}{\sqrt{2}} \tilde{\alpha}_0^{(K)} \quad \text{and} \quad \tilde{\alpha}_{-1}^{(K+1)} = \frac{1}{\sqrt{2}} \tilde{\alpha}_{-1}^{(K)}.$$

Similarly, after  $K + j$  total levels of decomposition we have

$$\tilde{\alpha}_k^{(K+j)} = \frac{1}{2^{j/2}} \tilde{\alpha}_k^{(K)} \leq \frac{1}{2^{j/2}}, \quad \text{for } k = -1, 0.$$

Thus,  $\|\tilde{\alpha}^{(K+j)}\|^2 = \left(\tilde{\alpha}_{-1}^{(K+j)}\right)^2 + \left(\tilde{\alpha}_0^{(K+j)}\right)^2 \leq 2^{-(j-1)}$ .

Let  $J = K + j$  where  $2^{-(j-1)} < \epsilon/2$ . Then  $\|\alpha^{(J)}\|^2 < \epsilon$  because  $\|\tilde{\alpha}^{(J)}\|^2 < \epsilon/2$  and  $\|\alpha^{(J)}\|^2$  cannot exceed  $\|\tilde{\alpha}^{(J)}\|^2$  by more than the energy  $\epsilon/2$  excluded in the truncation of  $x$ .

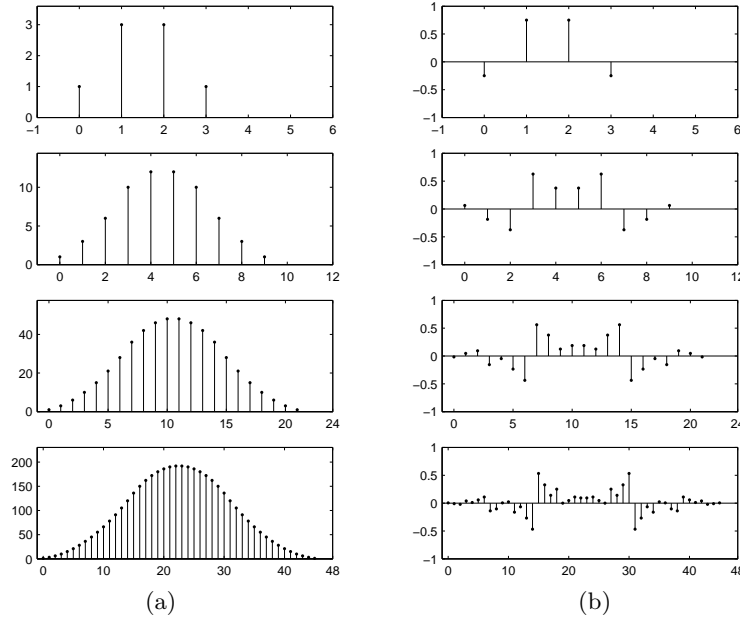
## 9.4 Biorthogonal Discrete Wavelet Transform

We have seen that the properties of the dyadic orthogonal DWT follow from the properties of the orthogonal two-channel filter bank. Similarly, the properties of the biorthogonal DWT follow from the properties of the biorthogonal two-channel filter bank. Instead of fully developing the biorthogonal DWT (as it is parallel to the orthogonal one), we quickly summarize its salient elements.

### 9.4.1 Definition of the Biorthogonal DWT

If, instead of an orthogonal pair of highpass/lowpass filters, we use a biorthogonal set  $\{h, g, \tilde{h}, \tilde{g}\}$  as in (7.64a)–(7.64d), we obtain two sets of equivalent filters, one for





**Figure 9.13:** Iteration of a biorthogonal pair of lowpass filters from (9.21): (a) the iteration of  $g_n$  leads to a *smooth*-looking sequence, while (b) the iteration of  $\tilde{g}$  does not.

the synthesis side,  $G^{(J)}, H^{(\ell)}$ ,  $\ell = 1, 2, \dots, J$ , and the other for the analysis side,  $\tilde{G}^{(J)}, \tilde{H}^{(\ell)}$ ,  $\ell = 1, 2, \dots, J$ :

$$G^{(J)}(z) = \prod_{k=0}^{J-1} G(z^{2^k}), \quad H^{(\ell)}(z) = H(z^{2^{\ell-1}})G^{(\ell-1)}(z), \quad (9.20a)$$

$$\tilde{G}^{(J)}(z) = \prod_{k=0}^{J-1} \tilde{G}(z^{2^k}), \quad \tilde{H}^{(\ell)}(z) = \tilde{H}(z^{2^{\ell-1}})\tilde{G}^{(\ell-1)}(z), \quad (9.20b)$$

for  $\ell = 1, \dots, J$ . This iterated product will play a crucial role in the construction of continuous-time wavelet bases in Chapter 12, as  $g^{(J)}$  and  $\tilde{g}^{(J)}$  can exhibit quite different behaviors; we illustrate this with an example.

**EXAMPLE 9.3 (BIORTHOGONAL DWT)** In Example 7.4, we derived a biorthogonal pair with lowpass filters

$$g_n = [\dots \ 0 \ 1 \ 3 \ 3 \ 1 \ 0 \ \dots], \quad (9.21a)$$

$$\tilde{g}_n = \frac{1}{4} [\dots \ 0 \ -1 \ 3 \ 3 \ -1 \ 0 \ \dots]. \quad (9.21b)$$

Figure 9.13 shows the first few iterations of  $g^{(J)}$  and  $\tilde{g}^{(J)}$  indicating a very different behavior. Recall that both filters are lowpass filters as are their iterated

versions. However, the iteration of  $\tilde{g}$  does not look *smooth*, indicating possible problems as we iterate to infinity (as we will see in Chapter 12).

Similarly to properties that equivalent filters satisfy in an orthogonal DWT (Section 9.2), we have such properties here mimicking the biorthogonal relations from Section 7.4; their formulation and proofs are left as an exercise to the reader.

**DEFINITION 9.3 (BIORTHOGONAL DWT)** The  $J$ -level biorthogonal DWT of a sequence  $x$  is a function of  $\ell \in \{1, 2, \dots, J\}$  given by

$$\alpha_k^{(J)} = \langle x_n, \tilde{g}_{n-2^J k}^{(J)} \rangle_n, \quad \beta_k^{(\ell)} = \langle x_n, \tilde{h}_{n-2^\ell k}^{(\ell)} \rangle_n, \quad k, n \in \mathbb{Z}, \ell \in \{1, 2, \dots, J\}. \quad (9.22a)$$

The inverse DWT is given by

$$x_n = \sum_{k \in \mathbb{Z}} \alpha_k^{(J)} g_{n-2^J k}^{(J)} + \sum_{\ell=1}^J \sum_{k \in \mathbb{Z}} \beta_k^{(\ell)} h_{n-2^\ell k}^{(\ell)}. \quad (9.22b)$$

In the above, the  $\alpha^{(J)}$  are the *scaling coefficients* and the  $\beta^{(\ell)}$  are the *wavelet coefficients*.

The equivalent filters  $g^{(J)}, \tilde{g}^{(J)}$  are often called the *scaling sequences*, and  $h^{(\ell)}, \tilde{h}^{(\ell)}, \ell = 1, 2, \dots, J$ , *wavelets* (wavelet sequences).

### 9.4.2 Properties of the Biorthogonal DWT

Similarly to the orthogonal DWT, the biorthogonal DWT is linear and shift varying. As a biorthogonal expansion, it does not satisfy Parseval's equality. However, as we have now access to dual bases, we can choose which one to use for projection (analysis) and which one for reconstruction (synthesis). This allows us to choose a better-suited one between  $g$  and  $\tilde{g}$  to induce an expansion with desired polynomial approximation properties and characterization of singularities.

## 9.5 Wavelet Packets

So far, the iterated decomposition was always applied to the lowpass filter, and often, there are good reasons to do so. However, to match a wide range of sequences, we can consider an arbitrary tree decomposition. In other words, start with a sequence  $x$  and decompose it into a lowpass and a highpass version.<sup>128</sup> Then, decide if the lowpass, the highpass, or both, are decomposed further, and keep going until a given depth  $J$ . The DWT is thus one particular case when only the lowpass version is repeatedly decomposed. Figure 9.7 depicts some of these decomposition possibilities.

<sup>128</sup>Of course, there is no reason why one could not split into  $N$  channels initially.

For example, the full tree yields a linear division of the spectrum similar to the local Fourier transform from Chapter 8, while the octave-band tree performs a  $J$ -level DWT expansion. Such arbitrary tree structures were introduced as a family of orthonormal bases for discrete-time sequences, and are known under the name of *wavelet packets*. The potential of wavelet packets lies in the capacity to offer a rich menu of orthonormal bases, from which the *best* one can be chosen (best according to a given criterion). We discuss this in more detail in Chapter 13.

What we do here is define the basis functions and write down the appropriate orthogonality relations; since the proofs are in principle similar to those for the DWT, we chose to omit them.

### 9.5.1 Definition of the Wavelet Packets

**Equivalent Channels and Their Properties** Denote the equivalent filters by  $g_{i,n}^{(\ell)}$ ,  $i = 0, \dots, 2^\ell - 1$ . In other words,  $g_i^{(\ell)}$  is the  $i$ th equivalent filter going through one of the possible paths of length  $\ell$ . The ordering is somewhat arbitrary, and we will choose the one corresponding to a full tree with a lowpass in the lower branch of each fork, and start numbering from the bottom.

**EXAMPLE 9.4 (2-LEVEL WAVELET PACKET EQUIVALENT FILTERS)** Let us find all equivalent filters at level 2, or, the filters corresponding to depth-1 and depth-2 trees.

$$\begin{aligned} G_0^{(1)}(z) &= G_0(z), & G_1^{(1)}(z) &= G_1(z), \\ G_0^{(2)}(z) &= G_0(z) G_0(z^2), & G_1^{(2)}(z) &= G_0(z) G_1(z^2), \\ G_2^{(2)}(z) &= G_1(z) G_0(z^2), & G_3^{(2)}(z) &= G_1(z) G_1(z^2). \end{aligned} \quad (9.23)$$

With the ordering chosen in the above equations for level 2, increasing index does not always correspond to increasing frequency. For ideal filters,  $G_2^{(2)}(e^{j\omega})$  chooses the range  $[3\pi/4, \pi)$ , while  $G_3^{(2)}(e^{j\omega})$  covers the range  $[\pi/2, 3\pi/4)$ . Beside the identity basis, which corresponds to the no-split situation, we have four possible orthonormal bases (full 2-level split, full 1-level split, full 1-level split plus either lowpass or highpass split).

**Wavelet Packet Bases** Among the myriad of possible bases wavelet packets generate, one can choose that one best fitting the sequence at hand.

**EXAMPLE 9.5 (2-LEVEL WAVELET PACKET BASES)** Continuing Example 9.4, we have a family  $W = \{\Phi_0, \Phi_1, \Phi_2, \Phi_3, \Phi_4\}$ , where  $\Phi_4$  is simply  $\{\delta_{n-k}\}_{k \in \mathbb{Z}}$ :

$$\Phi_0 = \{g_{0,n-2^2k}^{(2)}, g_{1,n-2^2k}^{(2)}, g_{2,n-2^2k}^{(2)}, g_{3,n-2^2k}^{(2)}\}_{k \in \mathbb{Z}}$$

corresponds to the full tree;

$$\Phi_1 = \{g_{1,n-2k}^{(1)}, g_{1,n-2^2k}^{(2)}, g_{0,n-2^2k}^{(2)}\}_{k \in \mathbb{Z}}$$

corresponds to the DWT tree;

$$\Phi_2 = \{g_{0,n-2k}^{(1)}, g_{2,n-2^2k}^{(2)}, g_{3,n-2^2k}^{(2)}\}_{k \in \mathbb{Z}}$$

corresponds to the tree with the highpass split twice; and,

$$\Phi_3 = \{g_{0,n-2k}^{(0)}, g_{1,n-2k}^{(1)}\}_{k \in \mathbb{Z}}$$

corresponds to the usual two-channel filter bank basis.

In general, we will have Fourier-like bases, given by

$$\Phi_0 = \{g_{0,n-2^Jk}^{(J)}, \dots, g_{2^J-1,n-2^Jk}^{(J)}\}_{k \in \mathbb{Z}}, \quad (9.24)$$

and wavelet-like bases, given by

$$\Phi_1 = \{g_{1,n-2k}^{(1)}, g_{1,n-2^2k}^{(2)}, \dots, g_{1,n-2^Jk}^{(J)}, g_{0,n-2^Jk}^{(J)}\}_{k \in \mathbb{Z}}. \quad (9.25)$$

That these are all bases follows trivially from each building block being a basis (either orthonormal or biorthogonal).

### 9.5.2 Properties of the Wavelet Packets

Exercises at the end of this chapter discuss various forms and properties of wavelet packets: biorthogonal wavelet packets in Exercise 9.4 and arbitrary wavelet packets in Exercise 9.4.

**Number of Wavelet Packets** How many wavelet packets (different trees) are there? Call  $N^{(J)}$  the number of trees of depth  $J$ , then we have the recursion

$$N^{(J)} = (N^{(J-1)})^2 + 1, \quad (9.26)$$

since each branch of the initial two-channel filter bank can have  $N^{(J-1)}$  possible trees attached to it and the +1 comes from not splitting at all. As an initial condition, we have  $N^{(1)} = 2$  (either no split or a single split). It can be shown that the recursion leads to an order of

$$N^{(J)} \sim 2^{2^J} \quad (9.27)$$

possible trees. Of course, many of these trees will be poor matches to real-life sequences, but an efficient search algorithm allowing to find the best match between a given sequence and a tree-structured expansion is possible. The proof of (9.26) is left as Exercise 9.3.

## 9.6 Computational Aspects

We now consider the computational complexity of the DWT, and show an elementary but astonishing result, in large part responsible for the popularity of the DWT: the complexity is linear in the input size.

**Complexity of the DWT** Computing a DWT amounts to computing a set of convolutions but with a twist crucial to the computational efficiency of the transform; as the decomposition progresses down the tree (see, for example, Figure 9.1), the sampling rate decreases. The implementation of  $J$ -level DWT for a length- $N$  signal with a filter bank is equivalent to a factorization

$$\begin{bmatrix} I_{(1-2^{-J+1})N} & 0 \\ 0 & \begin{bmatrix} H^{(J)} \\ G^{(J)} \end{bmatrix} \end{bmatrix} \cdots \begin{bmatrix} I_{3N/4} & 0 \\ 0 & \begin{bmatrix} H^{(3)} \\ G^{(3)} \end{bmatrix} \end{bmatrix} \begin{bmatrix} I_{N/2} & 0 \\ 0 & \begin{bmatrix} H^{(2)} \\ G^{(2)} \end{bmatrix} \end{bmatrix} \begin{bmatrix} H^{(1)} \\ G^{(1)} \end{bmatrix},$$

where  $H^{(\ell)}$ 's and  $G^{(\ell)}$  are the highpass and lowpass operators, respectively, each with downsampling by two (see (2.194) and (2.197) for  $\ell = 1$ ), both sparse.

In the DWT tree, the second level has similar cost to the first, but at half the sampling rate (see (2.268)). Continuing this argument, the cost of the  $J$ -level DWT is

$$L + \frac{L}{2} + \frac{L}{4} + \cdots + \frac{L}{2^{J-1}} < 2L$$

in both multiplications and additions, with the cost of the order of at most

$$C_{\text{DWT}} \sim 2NL \sim O(N), \quad (9.28)$$

that is, it is linear in the input size with a constant depending on the filter length.

While the cost remains bounded, the delay does not. If the first block contributes a delay  $D$ , the second will produce a delay  $2D$  and the  $\ell$ th block a delay  $2^{\ell-1}D$ , for a total delay of

$$D_{\text{DWT}} = D + 2D + 2^2D + \cdots + 2^{J-1}D = (2^J - 1)D.$$

This large delay is a serious drawback, especially for real-time applications such as speech coding.

**Complexity of the General Wavelet Packets** What happens for more general trees? Clearly, the worst case is for a full tree (see Figure 9.7(e)).

We start with a naive implementation of a  $2^J$ -channel filter bank, downsampled by  $2^J$ . Recall that, according to (9.5b), the length of the equivalent filters (in the DWT or the full tree) are of the order  $O(L2^J)$ . Computing each filter and downsampling by  $2^J$  leads to  $L$  operations per channel, or, for  $2^J$  channels we obtain

$$C_{\text{direct}} \sim NL2^J \sim O(NL2^J), \quad (9.29)$$

which grows exponentially with  $J$ . Exercise 9.5 compares these two implementations of the full tree with two Fourier-based ones, showing gains in computational cost.

As the sampling rate goes down, the number of channels goes up, and the two effects cancel each other. Therefore, for  $J$ , the cost amounts to

$$C_{\text{full}} \sim NLJ \sim O(NLJ), \quad (9.30)$$

multiplications or additions, again for a length- $N$  sequence and length- $L$  filter.

## Chapter at a Glance

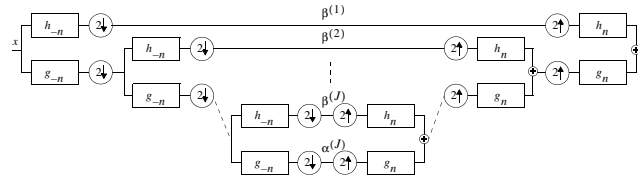
The goal of this chapter was twofold: (1) to extend the discussion from Chapters 7 and 8 to more multichannel filter banks constructed as trees and associated bases; and (2) to consider those filter banks implementing the DWT and wavelet packets.

While in general, tree-structured filter banks can have as their building blocks general  $N$ -channel filter banks, here we concentrated mostly on those built using basic two-channel filter banks. Moreover, the bulk of the chapter was devoted to those tree-structured filter banks, octave-band, where only the lowpass channel is further decomposed, as they implement the DWT. Such a decomposition is a natural one, with both theoretical and experimental evidence to support its use. Experimentally, research shows that the human visual system decomposes the field of view into octave bands; in parallel, theoretically, the DWT is an appropriate tool for the analysis of smooth sequences with isolated discontinuities. Moreover, the DWT has interesting polynomial approximation powers as well as the ability to characterize singularities.

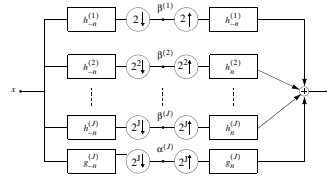
Wavelet packets extend these ideas to more general tilings of the time-frequency plane, adapting the decomposition to the sequence at hand. Here we discussed the decompositions only; the criteria for which decomposition to use are left for later.

### DWT Filter Bank

#### Block diagram: Tree structure



#### Block diagram: Multichannel structure



#### Basic characteristics

number of channels	$M = J + 1$
sampling at level $\ell$	$N^{(\ell)} = 2^\ell$
channel sequences	$\alpha_n^{(J)} \quad \beta_n^{(\ell)} \quad \ell = 1, 2, \dots, J$

#### Filters

#### Synthesis Analysis

	lowpass	bandpass <sup>(ℓ)</sup>	lowpass	bandpass <sup>(ℓ)</sup>	
orthogonal	$g_n^{(J)}$	$h_n^{(\ell)}$	$g_{-n}^{(J)}$	$h_{-n}^{(\ell)}$	$\ell = 1, 2, \dots, J$
biorthogonal	$g_n^{(J)}$	$h_n^{(\ell)}$	$\tilde{g}_n^{(J)}$	$\tilde{h}_n^{(\ell)}$	
polyphase component $j$	$g_{j,n}^{(J)}$	$h_{j,n}^{(\ell)}$	$\tilde{g}_{j,n}^{(J)}$	$\tilde{h}_{j,n}^{(\ell)}$	$j = 0, 1, \dots, 2^\ell - 1$

Table 9.3: DWT filter bank.

## Historical Remarks



The tree-structured filter banks, and, in particular, octave-band, or, constant-Q, ones, have been used in speech and audio. A similar scheme, but redundant, was proposed as a pyramid coding technique by Burt and Adelson [21], and served as the initial link between the discrete-time setting and the works in the continuous-time setting of Daubechies [39] and Mallat [99]. This prompted a flurry of connections between the wavelet transform, filter banks, and subband coding schemes, for example, the biorthogonal bases by Herley and Vetterli [71]. It, moreover, further opened the door to a formal treatment and definition of the DWT as a purely discrete transform, and not only as a vehicle for implementing continuous-time ones. Rioul [120] rigorously defined the discrete multiresolution analysis, and Coifman, Meyer, Quake and Wickerhauser [31] proposed wavelet packets as an adaptive tool for signal analysis. As a result of these developments and its low computational complexity, the DWT and its variations found their way into numerous applications and standards, JPEG 2000 among others.

## Further Reading

**Books and Textbooks** Numerous books cover the topic of the DWT, such as those by Vetterli and Kovačević [167], Strang and Nguyen [143] and Mallat [100], among others.

**Wavelet Packets** Wavelet packets were introduced in 1991 by Coifman, Meyer, Quake and Wickerhauser [31], followed by widely cited [172] and [173]. In 1993, Ramchandran and Vetterli showed for the first time a different cost measure for pruning wavelet packets, rate-distortion, as that suitable for compression [117], and Saito and Coifman extended the idea further with local discriminant bases [124, 123].

---

## Exercises with Solutions

### 9.7 Introduction

#### 9.1. Orthogonality Relations for the Orthogonal Tree-Structured Filter Bank

Given an orthogonal two-channel filter bank with filters  $g$  and  $h$ , prove the orthogonality relations (9.6a)–(9.6b), (9.11a)–(9.11d), as well as (9.14a)–(9.14b), for an octave-band filter bank, using similar arguments as in the proof of (7.13).

*Solution:* One part of the proof is demonstrating equivalences between the time-domain left-hand sides, and the right-hand sides involving matrix domain,  $z$ -transform domain and Fourier domain. We do this only for the simple case involving single filter, and then proceed to prove the orthogonality relations of the equivalent filters given orthogonality relations of the filters  $g$  and  $h$  in an orthogonal two-channel filter bank.

*Equivalences:* Demonstrating those expressions that involve single filter, such as (9.6a) and (9.11a), follows the same procedure as for  $N = 2$  in Section 2.7.5.

Instead of  $N = 2$ , in (9.6a) we have  $N = 2^J$ . Geometrically, the left-hand side of (9.6a) means that the columns of  $G^{(J)}U_{2^J}$ , the operator describing upsampling by  $2^J$

followed by filtering by  $G^{(J)}$ , are orthonormal to each other, that is,

$$I = (G^{(J)}U_{2^J})^T (G^{(J)}U_{2^J}) = U_{2^J}^T (G^{(J)})^T G^{(J)}U_{2^J} = D_{2^J} (G^{(J)})^T G^{(J)}U_{2^J}.$$

As before, we can see the left-hand side of (9.6a) as the deterministic autocorrelation of  $g^{(J)}$  downsampled by  $2^J$ . Write the deterministic autocorrelation of  $g^{(J)}$  as in (2.16),  $a_k^{(J)} = \langle g_n^{(J)}, g_{n-k}^{(J)} \rangle_n$ . Because of left-hand side of (9.6a), it has a single nonzero term modulo  $2^J$ ,  $g_0^{(J)} = 1$ ,  $a_{2^J k}^{(J)} = \delta_k$ .

Since we assume a real  $g^{(J)}$ , in the  $z$ -transform domain,  $A^{(J)}(z) = G^{(J)}(z)G^{(J)}(z^{-1})$  using (2.142). Using the orthogonality of the roots of unity (2.277c), we can accomplish keeping only the terms modulo  $2^J$  by adding  $A^{(J)}(W_{2^J}^k z)$  for  $k = 0, 1, \dots, 2^J - 1$ , and dividing by  $2^J$ . Therefore,  $a_{2^J k}^{(J)} = \delta_k$  can be expressed as

$$\sum_{k=0}^{2^J-1} A^{(J)}(W_{2^J}^k z) = \sum_{k=0}^{2^J-1} G^{(J)}(W_{2^J}^k z) G^{(J)}(W_{2^J}^{-k} z^{-1}) = 2^J,$$

which on the unit circle leads to

$$\sum_{k=0}^{2^J-1} \left| G^{(J)}(W_{2^J}^k e^{j\omega}) \right|^2 = 2^J,$$

the quadrature mirror formula generalized to  $N = 2^J$ .

The above discussion demonstrates equivalences in (9.6a), (9.6b), as well as (9.11a), (9.11c) with  $N = 2^\ell$  (instead of  $N = 2^J$ ).

*Orthogonality:* To demonstrate orthogonality, we prove orthogonality of: the lowpass filter (9.6a), an individual bandpass filter (9.11a) and different bandpass filters (9.11b), as well as the lowpass and bandpass filters (9.14a), given that (7.13), (7.14) and (7.22) hold.

- (i) We start with orthogonality of the lowpass filter (9.6a), which will also prove orthogonality of an individual bandpass filter (9.11a) (with substitution  $\ell$  for  $J$ ). This is equivalent to all the terms  $z^{2^J k}$ ,  $k \neq 0$ , in  $A^{(J)}(z)$  having zero coefficients. We write  $A^{(J)}(z)$  as:

$$\begin{aligned} A^{(J)}(z) &= G^{(J)}(z)G^{(J)}(z^{-1}) \\ &\stackrel{(a)}{=} [G^{(J-1)}(z)G(z^{2^{J-1}})] [G^{(J-1)}(z^{-1})G(z^{-2^{J-1}})] \\ &= [G(z^{2^{J-1}})G(z^{-2^{J-1}})] [G^{(J-1)}(z)G^{(J-1)}(z^{-1})] \\ &= A(z^{2^{J-1}})A^{(J-1)}(z), \end{aligned} \tag{E9.1-1}$$

where (a) follows from (9.5c), and  $A(z)$  is the deterministic autocorrelation of the lowpass filter  $g$ . This deterministic autocorrelation can also be expressed in the polyphase domain as:

$$A(z) = 1 + zA_1(z^2), \tag{E9.1-2}$$

with polyphase components  $A_0(z) = 1$  and  $A_1(z)$ .<sup>129</sup> Expressing  $A^{(J-1)}(z)$  in its polyphase form as well:

$$A^{(J-1)}(z) = 1 + \sum_{j=1}^{2^{J-1}-1} z^j A_j^{(J-1)}(z^{2^{J-1}}), \tag{E9.1-3}$$

<sup>129</sup>Because of the symmetry of  $A(z)$  in  $z$  and  $z^{-1}$ , we could have used either in front of the first polyphase component in (E9.1-2); we chose  $z$  for simplicity.



allows us to rewrite  $A^{(J)}(z)$  from (E9.1-1) as

$$\begin{aligned}
 A^{(J)}(z) &\stackrel{(a)}{=} A(z^{2^{J-1}}) A^{(J-1)}(z) \\
 &\stackrel{(b)}{=} [1 + z^{2^{J-1}} A_1(z^{2^J})] [1 + \sum_{j=1}^{2^{J-1}-1} z^j A_j^{(J-1)}(z^{2^{J-1}})] \\
 &= 1 + z^{2^{J-1}} A_1(z^{2^J}) + \sum_{j=1}^{2^{J-1}-1} z^j A_j^{(J-1)}(z^{2^{J-1}}) + \\
 &\quad z^{2^{J-1}} A_1(z^{2^J}) \sum_{j=1}^{2^{J-1}-1} z^j A_j^{(J-1)}(z^{2^{J-1}}), \tag{E9.1-4}
 \end{aligned}$$

where (a) follows from (E9.1-1), and (b) from (E9.1-2), (E9.1-3). The first two terms in (E9.1-4) clearly have no powers of  $z^{2^J k}$ . The third term contains terms which have exponents of the form:

$$\sum_{j=1}^{2^{J-1}-1} (j + m_j 2^{J-1}) = \sum_{j=1}^{2^{J-1}-1} j + \underbrace{2^{J-1} \sum_{j=0}^{2^{J-1}-1} m_j}_{\text{multiples of } 2^{J-1}},$$

where  $m_j \in \mathbb{Z}$ . The first sum is

$$\sum_{j=1}^{2^{J-1}-1} j = (2^{J-1} - 1 + 1)(2^{J-1} - 1)/2 = 2^{2J-3} - 2^{J-2},$$

with no powers of  $z^{2^J k}$ , and the second sum contains only powers of  $z^{2^{J-1}}$ . Therefore, the third term in (E9.1-4) does not contain powers of  $z^{2^J k}$ . The fourth term contains terms which have exponents of the form:

$$2^{J-1} + m_0 2^J + \sum_{j=1}^{2^{J-1}-1} (j + m_j 2^{J-1}) = m_0 2^J + \underbrace{2^{J-1} (1 + \sum_{j=1}^{2^{J-1}-1} m_j)}_{\text{multiples of } 2^{J-1}} + \sum_{j=1}^{2^{J-1}-1} j,$$

and does not contain powers of  $z^{2^J k}$  either.

- (ii) We now show orthogonality of different bandpass filters (9.11b). Without loss of generality, we assumed  $\ell < j$ . Now, orthogonality will hold if and only if in  $C^{(\ell,j)}(z)$  all the terms  $z^{2^\ell k}$  have zero coefficients.

The deterministic crosscorrelation  $C^{(\ell,j)}(z)$  can be written as

$$\begin{aligned}
 C^{(\ell,j)}(z) &= H^{(\ell)}(z) H^{(j)}(z^{-1}) \\
 &= [H(z^{2^{\ell-1}}) G^{(\ell-1)}(z)] [H(z^{-2^{j-1}}) G^{(j-1)}(z^{-1})] \\
 &= [H(z^{2^{\ell-1}}) G^{(\ell-1)}(z)] [H(z^{-2^{j-1}}) G(z^{-2^{j-2}}) \dots G(z^{-2^{\ell-1}}) G^{(\ell-1)}(z^{-1})] \\
 &= [H(z^{2^{\ell-1}}) G(z^{-2^{\ell-1}})] [H(z^{-2^{j-1}}) G(z^{-2^{j-2}}) \dots G(z^{-2^\ell})] [G^{(\ell-1)}(z) G^{(\ell-1)}(z^{-1})] \\
 &= C(z^{2^{\ell-1}}) [H(z^{-2^{j-1}}) G(z^{-2^{j-2}}) \dots G(z^{-2^\ell})] A^{(\ell-1)}(z).
 \end{aligned}$$

From the orthogonality of  $g$  and  $h$ , their deterministic crosscorrelation  $C(z) = G(z)H(z^{-1})$  has only one nonzero polyphase component,  $C_1(z)$ , and thus

$$C^{(\ell,j)}(z) = [z^{2^{\ell-1}} C_1(z^{2^\ell})] [H(z^{-2^{j-1}}) G(z^{-2^{j-2}}) \dots G(z^{-2^\ell})] [1 + \sum_{j=1}^{2^{\ell-1}-1} z^j A_j^{(\ell-1)}(z^{2^{\ell-1}})].$$

After expanding  $C^{(\ell,j)}(z)$ , the first term contains terms whose exponents are of the form

$$\begin{aligned} & 2^{\ell-1} + m_0 2^\ell + m_1 2^{j-1} + m_2 2^{j-2} + \dots + m_{j-\ell} 2^\ell \\ & = 2^{\ell-1} + 2^\ell (m_0 + m_1 2^{j-\ell-1} + m_2 2^{j-\ell-2} + \dots + m_{j-\ell}), \end{aligned}$$

for all  $m_0, m_1, \dots, m_{j-\ell} \in \mathbb{Z}$ , which cannot be multiples of  $2^{\ell k}$ . Similarly, the second term contains exponents of the form

$$\begin{aligned} & 2^{\ell-1} + m_0 2^\ell + m_1 2^{j-1} + m_2 2^{j-2} + \dots + m_{j-\ell} 2^\ell + \sum_{i=1}^{2^{\ell-1}-1} (i + 2^{\ell-1} n_i) \\ & = 2^{\ell-1} (1 + \sum_{i=1}^{2^{\ell-1}-1} n_i) + \sum_{i=1}^{2^{\ell-1}-1} i + 2^\ell (m_0 + m_1 2^{j-\ell-1} + m_2 2^{j-\ell-2} + \dots + m_{j-\ell}), \end{aligned}$$

for all  $m_0, m_1, \dots, m_{j-\ell}, n_1, \dots, n_{2^{\ell-1}-1} \in \mathbb{Z}$ , which cannot be multiples of  $2^{\ell k}$  either.

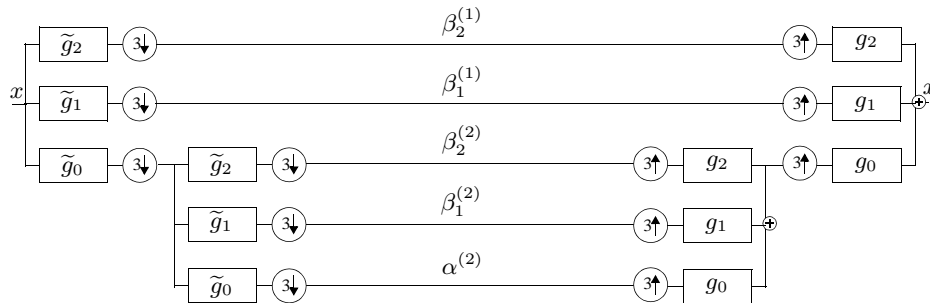
- (iii) We end by showing orthogonality of lowpass and bandpass filters (9.14a). This is equivalent to showing that in  $G^{(J)}(z)H^{(j)}(z^{-1})$ , all the terms  $z^{2^j}$  have zero coefficients. By rearranging terms, we get

$$G^{(J)}(z)H^{(j)}(z^{-1}) = \begin{cases} A^{(j-1)}(z)[G(z^{2^{j-1}})H(z^{-2^{j-1}})][G(z^{2^j}) \dots G(z^{2^{J-1}})] & j < J \\ A^{(j-1)}(z)[G(z^{2^{j-1}})H(z^{-2^{j-1}})] & j = J. \end{cases}$$

By substituting the expressions for  $A^{(j-1)}(z)$  and  $G(z^{2^{j-1}})H(z^{-2^{j-1}})$  from previous parts and expanding the terms, it is straightforward to verify that  $G^{(J)}(z)H^{(j)}(z^{-1})$  contains no powers of  $z^{2^j}$ .

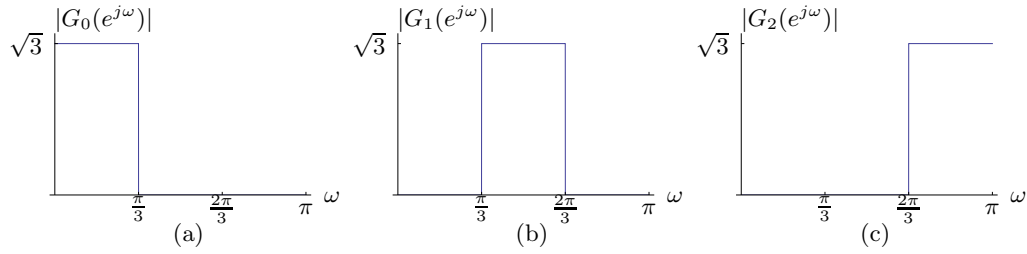
### 9.2. DWT from a 3-Channel Filter Bank

A 3-channel filter bank is iterated on the  $G_0$  branch to form a 2-level decomposition as shown in Figure E9.2-1.



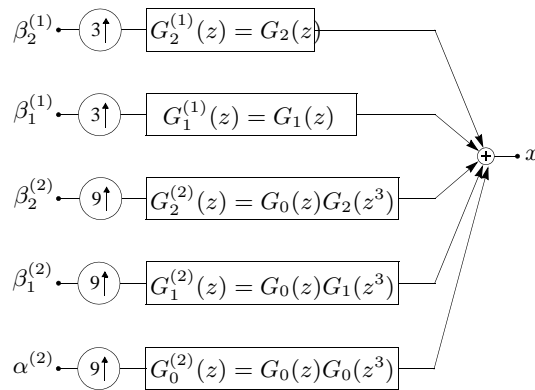
**Figure E9.2-1:** A 2-level DWT with 3-channel filter banks.

- Draw the equivalent 5-channel synthesis filter bank clearly specifying the transfer functions and sampling factors.
- Draw the time-frequency tiling of the filter bank.
- Write the basis matrix  $\Phi = \{\varphi_{k,n}\}_{k \in \mathbb{Z}}$ , for some 3-channel filter bank you specify (orthonormal or biorthogonal).
- If  $G$  are third-band ideal filters as shown in Figure E9.2-2, draw the magnitude responses of the equivalent filters.

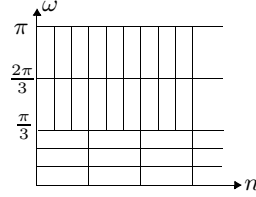
**Figure E9.2-2:** Third-band ideal filters.*Solution:*

- (i) Using the same process as in Section 9.2, the equivalent filters in a 5-channel filter bank are given in Figure E9.2-3 with:

$$\begin{aligned} G_2^{(1)}(z) &= G_2(z), & G_2^{(2)}(z) &= G_0(z)G_2(z^3), \\ G_1^{(1)}(z) &= G_1(z), & G_1^{(2)}(z) &= G_0(z)G_1(z^3), \\ & & G_0^{(2)}(z) &= G_0(z)G_0(z^3). \end{aligned}$$

**Figure E9.2-3:** Equivalent 5-channel synthesis filter bank corresponding to Figure E9.2-1.

- (ii) The time-frequency tiling corresponding to Figure E9.2-3 is given in Figure E9.2-4.
- (iii) Choose the following biorthogonal filter bank:  $G_0(z) = (1 + z^{-1} + z^{-2})/\sqrt{3}$ ,  $G_1(z) =$

**Figure E9.2-4:** Time-frequency tiling corresponding Figure E9.2-1.

$(1 - z^{-1} + z^{-2})/\sqrt{3}$ ,  $G_2(z) = (1 - z^{-1} - z^{-2})/\sqrt{3}$ . The synthesis filters are then

$$G_2^{(1)}(z) = \frac{1}{\sqrt{3}}(1 - z^{-1} - z^{-2})$$

$$G_1^{(1)}(z) = \frac{1}{\sqrt{3}}(1 - z^{-1} + z^{-2})$$

$$G_2^{(2)}(z) = \frac{1}{3}(1 + z^{-1} + z^{-2} - z^{-3} - z^{-4} - z^{-5} - z^{-6} - z^{-7} - z^{-8})$$

$$G_1^{(2)}(z) = \frac{1}{3}(1 + z^{-1} + z^{-2} - z^{-3} - z^{-4} - z^{-5} + z^{-6} + z^{-7} + z^{-8})$$

$$G_0^{(2)}(z) = \frac{1}{3}(1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + z^{-5} + z^{-6} + z^{-7} + z^{-8}),$$

and the basis matrix

$$\Phi = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 & \sqrt{3} & \sqrt{3} & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & -\sqrt{3} & -\sqrt{3} & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & -\sqrt{3} & \sqrt{3} & 0 & 0 & 0 & 0 \\ 1 & -1 & -1 & 0 & 0 & \sqrt{3} & \sqrt{3} & 0 & 0 \\ 1 & -1 & -1 & 0 & 0 & -\sqrt{3} & -\sqrt{3} & 0 & 0 \\ 1 & -1 & -1 & 0 & 0 & -\sqrt{3} & \sqrt{3} & 0 & 0 \\ 1 & 1 & -1 & 0 & 0 & 0 & 0 & \sqrt{3} & \sqrt{3} \\ 1 & 1 & -1 & 0 & 0 & 0 & 0 & -\sqrt{3} & -\sqrt{3} \\ 1 & 1 & -1 & 0 & 0 & 0 & 0 & -\sqrt{3} & \sqrt{3} \end{bmatrix}$$

(iv) The magnitude responses of the equivalent filters are given in Figure E9.2-5.

### 9.3. Biorthogonality Relations for the Biorthogonal DWT

Formulate and prove the appropriate biorthogonality relations corresponding to their orthogonal counterparts in (9.6a)–(9.6b), (9.11a)–(9.11c), as well as (9.14a)–(9.14b).

*Solution:* TBD.

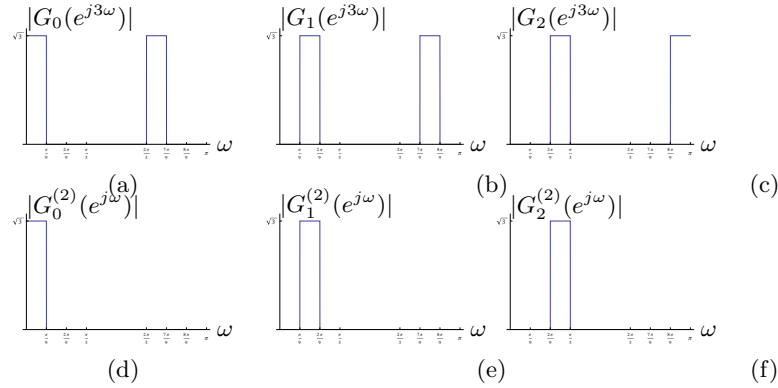
### 9.4. Wavelet Packets with Linear Phase Filters

Show that in a filter bank with linear-phase filters, the iterated filters are also linear phase. In particular, consider a two-channel filter bank with  $g_0$  and  $g_1$  of even length, symmetric and antisymmetric, respectively. Take an equivalent 4-channel filter bank, with  $G_0^{(2)}(z) = G_0(z)G_0(z^2)$ ,  $G_1^{(2)}(z) = G_0(z)G_1(z^2)$ ,  $G_2^{(2)}(z) = G_1(z)G_0(z^2)$ ,  $G_3^{(2)}(z) = G_1(z)G_1(z^2)$ , as in (9.23). What are the lengths and symmetries of these four filters?

*Solution:* A linear-phase filter satisfies (2.110), or, in  $z$ -domain,

$$H(z) = \pm z^{-L+1} H(z^{-1}). \quad (\text{E9.4-1})$$

In an iterated filter bank, filters at depth  $i$  have the form (9.5a) (where for wavelet packets, we have an arbitrary combination of lowpass and highpass filters). We now show that if  $g$



**Figure E9.2-5:** Magnitude responses of the equivalent filters corresponding to Figure E9.2-3 if the filters in the 3-channel filter bank are third-band ideal.

is linear phase, so is  $g^{(J)}$  (the proof would follow similarly for any other iterated filter):

$$\begin{aligned}
 G^{(J)}(z) &= \prod_{\ell=0}^{J-1} G(z^{2^\ell}) \stackrel{(a)}{=} \prod_{\ell=0}^{J-1} \pm (z^{2^\ell})^{-L+1} G(z^{-2^\ell}) \\
 &= (\pm 1)^\ell \prod_{\ell=0}^{J-1} (z^{2^\ell})^{-L+1} \prod_{\ell=0}^{J-1} G(z^{-2^\ell}) \stackrel{(b)}{=} (\pm 1)^\ell z^{(J-1) \sum_{\ell=0}^{J-1} 2^\ell} G^{(J)}(z^{-1}) \\
 &\stackrel{(c)}{=} (\pm 1)^\ell z^{(J-1)(2^J-1)} G^{(J)}(z^{-1}) \stackrel{(d)}{=} (\pm 1)^\ell z^{-L^{(J)}+1} G^{(J)}(z^{-1}),
 \end{aligned}$$

where (a) follows from each filter (in this case the same filter) being linear phase as in (E9.4-1); (b) from (9.5a); (c) from the geometric series formula (P1.65-1); and (d) from the expression for the total length of the iterated filter (9.5b). This shows that the equivalent iterated filter is linear phase as well.

The filters  $G_0(z)$  and  $G_1(z)$  are of even length, symmetric and antisymmetric, respectively. Then,  $G_0(z^2)$  and  $G_1(z^2)$  are of even length as well. Since the convolution of two even-length filters is of odd length, we know that all of the level-2 filters are of odd length.

Next,  $G_i(z^2)$  has the same symmetry properties as  $G_i(z)$  and thus:

- (i)  $G_0^{(2)}(z)$  is the product of two symmetric filters and is thus symmetric.
- (ii)  $G_1^{(2)}(z)$  is the product of a symmetric and an antisymmetric filter and is thus antisymmetric.
- (iii)  $G_2^{(2)}(z)$  is the product of an antisymmetric and a symmetric filter and is thus antisymmetric.
- (iv)  $G_3^{(2)}(z)$  is the product of two antisymmetric filters and is thus symmetric.

#### 9.5. Complexity of Computing a $J$ -Level, Full-Tree Filter Bank

Given is a full two-channel tree decomposition with  $J = \log_2(N)$  levels,  $N = 2^J$ , filters of length  $L$  at every stage and signals of length  $N$ . In (9.30) and (9.29), we gave two different expressions for the complexity of the full-tree implementation.

We now consider a couple of different options, both involving operations in the Fourier domain. Compute the overall complexity of the following two scenarios: First, for each, compute the DFT  $X_k$  of the input signal  $x_n$  using an FFT. Then:

- (i) Filter the input in the Fourier domain using  $2^J$  equivalent filters. (While we also need to take the DFT of these filters, that can be done in advance.)

- (ii) At each decomposition level of the tree, compute the Fourier transform  $A_k$  of the filtered and downsampled signal  $\alpha_n$  as

$$A_k = \frac{1}{2} \left( \tilde{G}_k X_k + \tilde{G}_{k+N/2} X_{k+N/2} \right)$$

for  $0 \leq k \leq N/2 - 1$ . The output signal  $A_k$  is of length  $N/2$ .

Repeat the procedure at every stage of the wavelet tree, but each time at half the sampling rate (thus halving the complexity of the previous stage).

Which of the four implementations we considered (two with time-domain filtering, (9.30) and (9.29), and two with Fourier-domain filtering, (i) and (ii)) has the lowest complexity? Does it depend on the number of stages  $J$ , filter length  $L$  and/or signal length  $N$ ?

*Solution:* The complexity of computing the FFT  $X_k$  of the input signal  $x_n$  is  $O(N \log N)$ .

- (i) The filtering operations can be performed as multiplications in Fourier domain. This requires  $N$  complex multiplications for each of the  $2^J$  filters, because they should have the same length as the input signal. Next, the filtered signals have to be downsampled. Downsampling by 2 can be evaluated as

$$A_k = \frac{1}{2} (X_k + X_{k+N/2}), \quad 0 \leq k \leq N/2 - 1,$$

and requires  $N/2$  additions. Similarly, downsampling by  $2^\ell$  can be evaluated as

$$A_k = \frac{1}{2^\ell} \sum_{\ell=0}^{2^\ell} X_{k+\ell N/2^\ell}, \quad 0 \leq k \leq \frac{N}{2^\ell} - 1,$$

and requires  $(2^\ell - 1)N/2^\ell$  additions. On the synthesis side, upsampling does not require any special operations, synthesis filtering is similar to the analysis filtering and the inverse FFT has again complexity  $O(N \log N)$ , resulting in the overall complexity of

$$C_a = N \log N + 2^J N + 2^J N - 2N + 2^J N + N \log N = 2N \log N + 3N^2.$$

- (ii) The computation of  $A_k$  as

$$A_k = \frac{1}{2} \left( \tilde{G}_k X_k + \tilde{G}_{k+N/2} X_{k+N/2} \right) \quad 0 \leq k \leq N/2 - 1,$$

requires  $N$  multiplications and  $N/2$  additions. When we repeat this at the different decomposition levels, we need

$$N + \frac{N}{2} + \frac{N}{4} + \dots < 2N$$

operations (the synthesis side has similar complexity). The overall complexity is thus

$$C_b = 2(N \log N + 2N) = 2N \log N + 4N.$$

Both algorithms from (i), (ii) have lower complexity than their counterparts from (9.30), (9.29). Which is less complex depends on the signal and filter lengths. They have the same complexity for

$$4NL = 4N + 2N \log N \Rightarrow L = 1 + \frac{\log N}{2}.$$

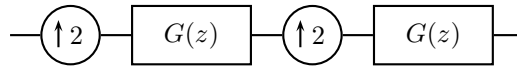
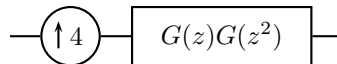
With shorter filter lengths, the time-domain algorithm has lower complexity. With filter lengths longer than  $1 + \log N/2$ , the frequency-domain algorithm is more efficient.

## Exercises

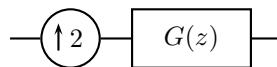
### 9.8 Introduction

#### 9.1. Iteration of an Upsampled Interpolation Filter

Given is the system as in Figure P9.1-1 where  $G(z)$  is the following interpolation filter  $G(z) = \frac{1}{2}z + 1 + \frac{1}{2}z^{-1}$ .

**Figure P9.1-1:** System for Exercise 9.1.**Figure P9.1-2:** System equivalent to that of Figure P9.1-1.

- (i) Prove that the system in Figure P9.1-1 is equivalent to that of Figure P9.1-2.
- (ii) Iterate the system in Figure P9.1-3  $N$ -times and give the equivalent filter in the  $z$ -transform domain.

**Figure P9.1-3:** System equivalent to that of Figure P9.1-1.

- (iii) For  $N = 2$ , sketch the impulse response of the equivalent filter from (ii) and explain why it is called a linear interpolator.

**9.2. Equivalent Filters**

Given the expression for the equivalent filter as in (9.5a), prove the following recursive relations:

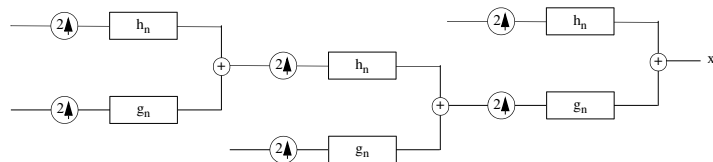
- (i)  $G^{(J)}(z) = G(z) G^{(J-1)}(z^2)$ .
- (ii)  $G^{(J)}(z) = G(z^{2^{J-1}}) G^{(J-1)}(z)$ .
- (iii)  $G^{(2^k)}(z) = G^{(2^{k-1})}(z) G^{(2^{k-1})}(z^{2^{2^{k-1}}})$ .

**9.3. Number of Wavelet Packets**

Prove that the number of possible wavelet packet bases generated from a depth- $L$  binary tree is as given in (9.26).

**9.4. Wavelet Packets**

Given is the following synthesis wavelet packet tree:

**Figure P9.4-1:** An arbitrary wavelet packet tree.

- (i) Draw the equivalent time-frequency tiling induced by such a wavelet packet tree.
- (ii) Identify the equivalent 4-channel synthesis filter bank and compute the equivalent filters. You may assume that the filters in each two-channel filter bank are Haar from Table 7.8, that is,  $G(z) = (1 + z^{-1})/\sqrt{2}$  and  $H(z) = (1 - z^{-1})/\sqrt{2}$ .
- (iii) Identify the basis functions of such an expansion. Does it implement an orthonormal basis? Why?

#### 9.5. Orthogonal Full-Tree Filter Banks

Given is a full-tree filter bank of depth 2.

- (i) Assume ideal sinc filters, and give the frequency response magnitude of  $G_i^{(2)}(e^{j\omega})$ ,  $i = 0, 1, 2, 3$ . Note that this is not the natural ordering one would expect.
- (ii) Now take the Haar filters, and give  $g_i^{(2)}$ ,  $i = 0, 1, 2, 3$ . These are the discrete-time Walsh-Hadamard functions of length 4.
- (iii) Given that  $\{g_0, g_1\}$  is an orthogonal pair, prove orthogonality for any of the equivalent filters with respect to shifts by 4.

#### 9.6. 6-Channel Filter Bank

Given a filter bank specified by the following subband signal equations:

$$\begin{aligned}
 x^{(1)} &= D_2 G D_2 G x, \\
 x^{(2)} &= D_2 G D_2 H D_2 G x, \\
 x^{(3)} &= D_2 H D_2 H D_2 G x, \\
 x^{(4)} &= D_2 G D_2 G D_2 H x, \\
 x^{(5)} &= D_2 H D_2 G D_2 H x, \\
 x^{(6)} &= D_2 H D_2 H x,
 \end{aligned}$$

where  $G$  and  $H$  are the lowpass, highpass filter operators, respectively.

- (i) Draw the block diagram of the system using two-channel filter banks.
- (ii) Draw the equivalent time-frequency tiling induced by the 6-channel filter bank.
- (iii) Draw the equivalent single-level 6-channel filter bank clearly specifying the down-sampling factors and the equivalent filters in each branch.



## Chapter 10

# Local Fourier and Wavelet Frames on Sequences

## Contents

10.1	Introduction . . . . .	714
10.2	Finite-Dimensional Frames . . . . .	725
10.3	Oversampled Filter Banks . . . . .	744
10.4	Local Fourier Frames . . . . .	750
10.5	Wavelet Frames . . . . .	757
10.6	Computational Aspects . . . . .	763
	Chapter at a Glance . . . . .	765
	Historical Remarks . . . . .	766
	Further Reading . . . . .	766
	Exercises with Solutions . . . . .	768
	Exercises . . . . .	772

Redundancy is a common tool in our daily lives; it helps remove doubt or uncertainty. Redundant signal representations follow the same idea to create robustness. Given a sequence, we often represent it in another domain where its characteristics are more readily apparent in the expansion coefficients. If the representation in that other domain is achieved via a basis, corruption or loss of expansion coefficients can be serious. If, on the other hand, that representation is achieved via a redundant representation, such problems can be avoided.

As introduced in Chapter 1, Section 1.5.4, the redundant counterpart of bases are called *frames*. Frames are the topic of the present chapter. The building blocks of a representation can be seen as words in a dictionary; while a basis uses a minimum number of such words, a frame uses an overcomplete set. This is similar to multiple words with slight variations for similar concepts, allowing for very short sentences describing complex ideas.<sup>130</sup> While in most of the previous chapters, our emphasis was on finding the *best* expansion/representation vectors (Fourier, wavelet, etc.), frames allow us even more freedom; not only do we look for the *best*

<sup>130</sup>As the urban legend goes, Eskimos have a hundred words for snow.

*expansion*, we can also look for the *best expansion coefficients* given a fixed expansion and under desired constraints (sparsity as one example). This freedom is due to the fact that in a redundant dictionary, the expansion coefficients are not unique, while in a basis they are.

We briefly introduced frames in  $\mathbb{R}^2$  in Chapter 1, Section 1.1. We followed this by a more formal discussion in Section 1.5.4. Our goal in this chapter is to explore the potential of such overcomplete (redundant) representations in specific settings; in particular, we study fundamental properties of frames, both in finite dimensions as well as in  $\ell^2(\mathbb{Z})$ . We look into designing frames with some structure, especially those implementable by oversampled filter banks, and more specifically those with Fourier-like or wavelet-like time-frequency behavior. We end the chapter with the discussion of computational aspects related to frame expansions.

*Notation used in this chapter:* We consider both real-coefficient and complex-coefficient frames here, unlike, for example, in Chapter 7. When Hermitian conjugation is applied to polyphase matrices, it is applied only to coefficients and not to  $z$ .  $\square$

## 10.1 Introduction

Redundant sets of vectors look like an *overcomplete* basis,<sup>131</sup> and we call such sets frames. Thus, a frame is an extension of a basis, where, for a given space, more vectors than necessary are used to obtain an expansion with desirable properties. In this section, we use the two frame examples from Chapter 1, Section 1.1, to introduce and discuss frame concepts in a simple setting but in more detail. For ease of presentation, we will repeat pertinent equations as well as figures.

### A Tight Frame for $\mathbb{R}^2$

Our first example is that from (1.15), a set of three vectors in  $\mathbb{R}^2$ ,

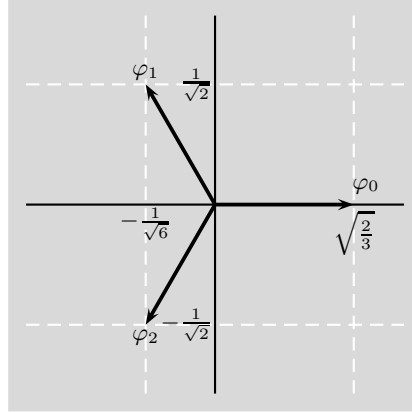
$$\varphi_0 = \begin{bmatrix} \sqrt{\frac{2}{3}} \\ 0 \end{bmatrix}, \quad \varphi_1 = \begin{bmatrix} -\frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}, \quad \varphi_2 = \begin{bmatrix} -\frac{1}{\sqrt{6}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix}. \quad (10.1)$$

**Expansion** These vectors clearly span  $\mathbb{R}^2$  since any two of them do (see Figure 10.1). How do we represent a vector  $x \in \mathbb{R}^2$  as a linear combination of  $\{\varphi_i\}_{i=0,1,2}$ ,

$$x = \sum_{i=0}^2 \alpha_i \varphi_i? \quad (10.2)$$

Let us gather these vectors into a frame matrix  $\Phi$ ,

<sup>131</sup>Even though this term is a contradiction.



**Figure 10.1:** The three vectors  $\{\varphi_0, \varphi_1, \varphi_2\}$  from (10.1) form a frame for  $\mathbb{R}^2$  (the same as Figure 1.4(b)).

$$\Phi = [\varphi_0 \ \varphi_1 \ \varphi_2] = \begin{bmatrix} \sqrt{\frac{2}{3}} & -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{6}} \\ 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}. \quad (10.3)$$

To compute the expansion coefficients  $\alpha_i$  in (10.2), we need a right inverse of  $\Phi$ . Since  $\Phi$  is rectangular, such an inverse is not unique, so we look for the simplest one. As the rows of  $\Phi$  are orthonormal,  $\Phi \Phi^T = I_2$ , and thus, a possible right inverse of  $\Phi$  is just its transpose:

$$\Phi^T = \begin{bmatrix} \sqrt{\frac{2}{3}} & 0 \\ -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} \varphi_0^T \\ \varphi_1^T \\ \varphi_2^T \end{bmatrix}. \quad (10.4)$$

Gathering the expansion coefficients into a vector  $\alpha$ ,

$$\alpha = \Phi^T x, \quad (10.5)$$

and, using the fact that

$$\Phi \alpha = \Phi \Phi^T x = x,$$

we obtain the following expansion formula:

$$x = \sum_{i=0}^2 \langle x, \varphi_i \rangle \varphi_i, \quad (10.6)$$

for all  $x \in \mathbb{R}^2$ , which looks exactly like the usual orthonormal expansion (1.85a), except for the number of vectors involved.

**Geometry of the Expansion** Let us understand why this is so. The rows of  $\Phi$  are orthonormal,

$$\Phi\Phi^T = I_2. \quad (10.7)$$

Actually,  $\Phi$  can be seen as two rows of a unitary matrix whose third row is orthogonal to the rows of  $\Phi$ . We call that third row  $\Phi^\perp$ , that is,  $\Phi^\perp = 1/\sqrt{3} [1 \ 1 \ 1]$ . That unitary matrix is then

$$\begin{bmatrix} \Phi \\ \Phi^\perp \end{bmatrix} = \begin{bmatrix} \sqrt{\frac{2}{3}} & -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{6}} \\ 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{bmatrix}. \quad (10.8)$$

We can then write

$$\Phi\Phi^T = I_{2 \times 2}, \quad (10.9a)$$

$$\Phi(\Phi^\perp)^T = 0_{2 \times 1}, \quad (10.9b)$$

$$\Phi^\perp(\Phi^\perp)^T = I_{1 \times 1} = 1. \quad (10.9c)$$

Calling  $S$  the subspace of  $\mathbb{R}^3$  spanned by the columns of  $\Phi^T$ , and  $S^\perp$  its orthogonal complement in  $\mathbb{R}^3$  (spanned by the one column of  $(\Phi^\perp)^T$ ), we can write

$$\begin{aligned} S &= \text{span}(\Phi^T) = \text{span} \left( \begin{bmatrix} \sqrt{\frac{2}{3}} \\ -\frac{1}{\sqrt{6}} \\ -\frac{1}{\sqrt{6}} \end{bmatrix}, \begin{bmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} \right), \\ S^\perp &= \text{span}((\Phi^\perp)^T) = \text{span} \left( \begin{bmatrix} \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \end{bmatrix} \right), \\ \mathbb{R}^3 &= S \oplus S^\perp. \end{aligned}$$

We just saw that the rows of  $\Phi$  are orthonormal; moreover, while not of unit norm, the columns of  $\Phi$  are of the same norm,  $\|\varphi_i\| = \sqrt{2/3}$ . Therefore,  $\Phi$  is a very special matrix, as can be guessed by looking at Figure 10.1.

Let us now understand the nature of the expansion coefficients  $\alpha$  a bit more in depth. Obviously,  $\alpha$  cannot be arbitrary; since  $\alpha = \Phi^T x$ , it belongs to the range of the columns of  $\Phi^T$ , or,  $\alpha \in S$ . What about some arbitrary  $\alpha' \in \mathbb{R}^3$ ? As the expansion coefficients must belong to  $S$ , we can calculate the orthogonal projection of  $\alpha'$  onto  $S$  by first calculating some  $x' = \Phi\alpha'$ , and then computing the unique orthogonal projection we call  $\alpha$  as

$$\alpha = \Phi^T x' = \Phi^T \Phi \alpha',$$

where  $G = \Phi^T \Phi$  is the Gram matrix from (1.112),

$$G = \Phi^T \Phi \stackrel{(a)}{=} \frac{1}{3} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}, \quad (10.10)$$

and (a) follows from (10.3), (10.4). We can therefore express any  $\alpha' \in \mathbb{R}^3$  as

$$\alpha' = \alpha + \alpha^\perp \quad (10.11a)$$

with  $\alpha \in S$  and  $\alpha^\perp \in S^\perp$ , and thus

$$\langle \alpha, \alpha^\perp \rangle = 0. \quad (10.11b)$$

Therefore, we see that, for our  $\Phi$ , many different  $\alpha'$  are possible as expansion coefficients. In other words,  $\Phi^T$  is not the only possible right inverse of  $\Phi$ . While this is not surprising, it allows frames to be extremely flexible, a fact we will explore in detail in the next section. Throughout this chapter, when we write  $\alpha'$ , we will mean any vector of expansion coefficients; in contrast,  $\alpha$  will be the unique one obtained using the canonical (unique) dual frame.

**Energy of Expansion Coefficients** For orthonormal bases, Parseval's equality (energy conservation) (1.87a) is fundamental. To find out what happens here, we compute the norm of  $\alpha$ ,

$$\|\alpha\|^2 = \alpha^T \alpha \stackrel{(a)}{=} x^T \Phi \Phi^T x \stackrel{(b)}{=} x^T x = \|x\|^2, \quad (10.12)$$

where (a) follows from (10.5); and (b) from (10.7), again formally the same as for an orthonormal basis. Beware though that the comparison is not entirely fair, as the frame vectors are not of unit norm; we will see in a moment what happens when this is the case.

**Robustness to Corruption and Loss** What does the redundancy of this expansion buy us? For example, what if the expansion coefficients get corrupted by noise? Assume, for instance, that  $\alpha$  is perturbed by noise  $\eta'$ , where the noise components  $\eta'_i$  are uncorrelated with  $\|\eta'\| = 1$ . Then, reconstruction will project the noise  $\eta' = \eta + \eta^\perp$ , and thus cancel that part of  $\eta$  not in  $S$ :

$$y = \Phi(\alpha' + \eta') = x + \underbrace{\Phi\eta}_{x_\eta} + \underbrace{\Phi\eta^\perp}_0.$$

To compute  $\|x_\eta\|^2$ , we write

$$\|x_\eta\|^2 = x_\eta^T x_\eta = \eta^T \Phi^T \Phi \eta = \eta^T U \Sigma U^T \eta,$$

where we have performed a singular value decomposition (1.213) on  $G = \Phi^T \Phi$  as

$$G = \Phi^T \Phi = U \Sigma U^T = \begin{bmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{3}} \\ 0 & \sqrt{\frac{2}{3}} & \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} 1 & & \\ & 1 & \\ & & 0 \end{bmatrix} \begin{bmatrix} -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{6}} & \sqrt{\frac{2}{3}} & -\frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{bmatrix},$$

and  $U$  is a unitary matrix. So  $\|\eta'^T U\| = 1$ , and thus  $\|x_\eta\|^2 = (2/3)\|\eta\|^2$ . We have thus established that the energy of the noise gets reduced during reconstruction by the *contraction* factor  $2/3$ .

We have just looked at the effect of noise on the reconstructed vector in a frame expansion. We now ask a different question: What happens if, for some reason, we have access to only two out of three expansion coefficients during reconstruction (for example, one was lost)? As in this case, any two remaining vectors form a basis, we will still be able to reconstruct; however, the reconstruction is now performed differently. For example, assume we lost the first expansion coefficient  $\alpha_0$ . To reconstruct, we must behave as if we had started without the first vector  $\varphi_0$  and had computed the expansion coefficients using only  $\varphi_1$  and  $\varphi_2$ . This further means that to reconstruct, we must find the inverse of the  $2 \times 2$  submatrix of  $\Phi^T$  formed by taking its last two rows. This new reconstruction matrix  $\Phi^e$  (where  $e$  stands for *erasures*) is

$$\Phi^e = \begin{bmatrix} -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{2}} \end{bmatrix}^{-1} = \begin{bmatrix} -\frac{\sqrt{3}}{\sqrt{2}} & -\frac{\sqrt{3}}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix},$$

and thus, multiplying  $[\alpha_1 \ \alpha_2]^T$  by  $\Phi^e$  reconstructs the input vector:

$$\Phi^e \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} -\frac{\sqrt{3}}{\sqrt{2}} & -\frac{\sqrt{3}}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}.$$

**Unit-Norm Version** The frame we have just seen is a very particular frame and intuitively *close* to an orthonormal basis. However, there is one difference: while all the frame vectors are of the same norm  $\sqrt{2/3}$ , they are not of unit norm. We can normalize  $\|\varphi_i\|$  to be of norm 1, leading to

$$\Phi = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}, \quad \Phi^T = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}, \quad (10.13)$$

yielding the expansion

$$x = \frac{2}{3} \sum_{i=0}^2 \langle x, \varphi_i \rangle \varphi_i, \quad (10.14)$$

and the energy in the expansion coefficients

$$\|\alpha\|^2 = \frac{3}{2} \|x\|^2. \quad (10.15)$$

The difference between (10.13) and (10.3) is in normalization; thus, the factor  $(3/2)$  appears in (10.15), showing that the energy in the expansion coefficients is  $(3/2)$  times larger than that of the input vector. When the frame vectors are of unit norm as in this case, this factor represents the *redundancy*<sup>132</sup> of the system—we have  $(3/2)$  times more vectors than needed to represent a vector in  $\mathbb{R}^2$ .

The frame (10.3) and its normalized version (10.13) are instances of the so-called *tight frames*. A tight frame has a right inverse that is its own transpose and

<sup>132</sup>There exists a precise quantitative definition of redundancy, see *Further Reading* for details.

conserves energy (both within a scale factor). While the tight frame vectors we have seen are of the same norm, in general, this is not a requirement for tightness, as we will see later in the chapter.

**Filter-Bank Implementation** As we have seen in previous chapters, infinite-dimensional expansions can be implemented using filter banks. Let us for a moment go back to the Haar expansion for  $\ell^2(\mathbb{Z})$  and try to draw some parallels. First, we have, until now, seen many times a  $2 \times 2$  Haar matrix  $\Phi$ , as a basis for  $\mathbb{R}^2$ . Then, in Chapter 6, we used these Haar vectors to form a basis for  $\ell^2(\mathbb{Z})$ , by slicing the infinite-length sequence into pieces of length 2 and applying a Haar basis to each of these. The resulting basis sequences for  $\ell^2(\mathbb{Z})$  are then obtained as infinite sequences with two nonzero elements only, shifted by integer multiples of 2, (I.3), (I.5). Finally, in Chapter 7, (7.2)–(7.2), we showed how to implement such an orthonormal expansion for  $\ell^2(\mathbb{Z})$  by using a two-channel filter bank.

We can do exactly the same here. We slice an infinite-length input sequence into pieces of length 2 and apply the frame we just saw to each of these. To form a frame for  $\ell^2(\mathbb{Z})$ , we form three template frame vectors from the vectors (10.1) as:

$$\varphi_0 = \begin{bmatrix} \vdots \\ 0 \\ \boxed{\sqrt{\frac{2}{3}}} \\ 0 \\ 0 \\ \vdots \end{bmatrix}, \quad \varphi_1 = \begin{bmatrix} \vdots \\ 0 \\ \boxed{-\frac{1}{\sqrt{6}}} \\ \frac{1}{\sqrt{2}} \\ 0 \\ \vdots \end{bmatrix}, \quad \varphi_2 = \begin{bmatrix} \vdots \\ 0 \\ \boxed{-\frac{1}{\sqrt{6}}} \\ -\frac{1}{\sqrt{2}} \\ 0 \\ \vdots \end{bmatrix}. \quad (10.16)$$

We then form all the other frame sequences as versions of (10.16) shifted by integer multiples of 2:

$$\Phi = \{\varphi_{0,n-2k}, \varphi_{1,n-2k}, \varphi_{2,n-2k}\}_{k \in \mathbb{Z}}.$$

To implement this frame expansion using signal processing machinery, we do exactly the same as we did for Haar basis in Section 7.1: we rename the template frame sequences  $\varphi_0 = g_0$ ,  $\varphi_1 = g_1$  and  $\varphi_2 = g_2$ . Then we can write the reconstruction formula as

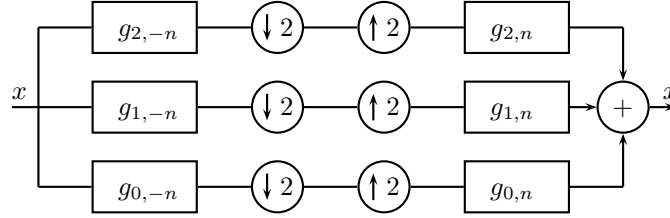
$$x_n = \sum_{k \in \mathbb{Z}} \alpha_{0,k} g_{0,n-2k} + \sum_{k \in \mathbb{Z}} \alpha_{1,k} g_{1,n-2k} + \sum_{k \in \mathbb{Z}} \alpha_{2,k} g_{2,n-2k}, \quad (10.17a)$$

with

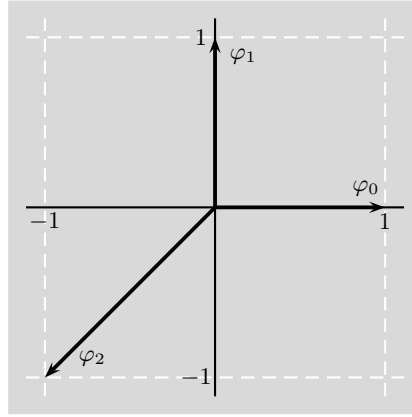
$$\alpha_{i,k} = \langle x_n, g_{i,n-2k} \rangle_n. \quad (10.17b)$$

There is really no difference between (10.17a)–(10.17b) and (7.2)–(7.3), except that we have 3 template frame sequences here instead of 2 template basis sequences for Haar.<sup>133</sup> We thus know exactly how to implement (10.17): it is going to be a 3-channel filter bank with down/upsampling by 2, as shown in Figure 10.2, with synthesis filters' impulse responses given by the frame vectors, and analysis filters' impulse responses given by the time-reversed frame vectors.

<sup>133</sup>Unlike for the Haar case, the  $\alpha_i$  are not unique.



**Figure 10.2:** A 3-channel filter bank with sampling by 2 implementing a tight frame expansion.



**Figure 10.3:** The three vectors  $\{\varphi_0, \varphi_1, \varphi_2\}$  from (10.18) form a frame for  $\mathbb{R}^2$  (the same as Figure 1.4(a)).

### A General Frame for $\mathbb{R}^2$

Our second example is that from (1.14), again a set of three vectors in  $\mathbb{R}^2$ ,

$$\varphi_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \varphi_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \varphi_2 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \quad (10.18)$$

the standard orthonormal basis  $\{\varphi_0, \varphi_1\}$  plus a third vector. We follow the same path as we just did to spot commonalities and differences.

**Expansion** Again, these vectors clearly span  $\mathbb{R}^2$  since any two of them do (see Figure 10.3). We have already paved the path for representing a vector  $x \in \mathbb{R}^2$  as a linear combination of  $\{\varphi_i\}_{i=0,1,2}$  by introducing a matrix  $\Phi$  as in (1.16a),

$$\Phi = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix}. \quad (10.19)$$



Unlike (10.3), this  $\Phi$  does not have orthogonal rows, and thus,  $\Phi^T$  is not one of its right inverses. We call these right inverses  $\tilde{\Phi}^T$  and  $\tilde{\Phi}$  a *dual frame*. Then

$$\Phi \tilde{\Phi}^T = I_2. \quad (10.20)$$

We have seen one possible dual frame in (1.16c),<sup>134</sup>

$$\tilde{\Phi} = \begin{bmatrix} 0 & -1 & -1 \\ -1 & 0 & -1 \end{bmatrix},$$

with the associated expansion

$$x \stackrel{(a)}{=} \sum_{i=0}^2 \langle x, \tilde{\varphi}_i \rangle \varphi_i \stackrel{(b)}{=} \sum_{i=0}^2 \langle x, \varphi_i \rangle \tilde{\varphi}_i, \quad (10.21)$$

where we expressed  $x$  both in the frame (a) and the dual frame (b). This looks exactly like the usual biorthogonal expansion (1.105a), except for the number of vectors involved. The dual frame vectors are:

$$\tilde{\varphi}_0 = \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \quad \tilde{\varphi}_1 = \begin{bmatrix} -1 \\ 0 \end{bmatrix}, \quad \tilde{\varphi}_2 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}. \quad (10.22)$$

**Geometry of the Expansion** In the previous example, the geometry of the expansion was captured by  $\Phi$  and  $\Phi^\perp$  as in (10.8); here, we must add the dual frame  $\tilde{\Phi}$  and its complement  $\tilde{\Phi}^\perp$ . Two possible complements corresponding to  $\Phi$  and  $\tilde{\Phi}$  are

$$\Phi^\perp = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}, \quad \tilde{\Phi}^\perp = \begin{bmatrix} 1 & 1 & -1 \end{bmatrix} \quad (10.23)$$

both of size  $1 \times 3$ . Then, the following capture the geometry of the matrices involved:

$$\Phi \tilde{\Phi}^T = I_{2 \times 2}, \quad (10.24a)$$

$$\Phi (\tilde{\Phi}^\perp)^T = 0_{2 \times 1}, \quad (10.24b)$$

$$\Phi^\perp \tilde{\Phi}^T = 0_{1 \times 2}, \quad (10.24c)$$

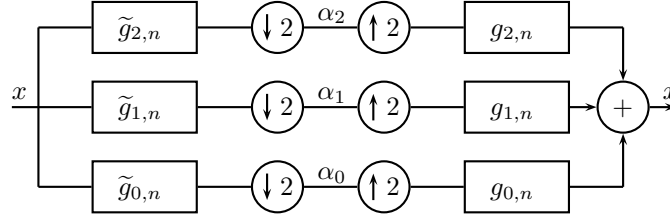
$$\Phi^\perp (\tilde{\Phi}^\perp)^T = I_{1 \times 1} = 1. \quad (10.24d)$$

Thus,  $\mathbb{R}^3$  is spanned by both  $\Phi^T \oplus (\Phi^\perp)^T$  and  $\tilde{\Phi}^T \oplus (\tilde{\Phi}^\perp)^T$ .

**Energy of Expansion Coefficients** We saw how energy is conserved in a Parseval-like manner before, what can we say about  $\|\alpha\|$  here?

$$\|\alpha\|^2 = \alpha^T \alpha = x^T \tilde{\Phi} \tilde{\Phi}^T x = x^T U \Sigma U^T x,$$

<sup>134</sup>Note that while there exist infinitely many dual frames since  $\Phi$  has a nontrivial null space, here we concentrate on the canonical one as will be clear later in the chapter.



**Figure 10.4:** A 3-channel filter bank with sampling by 2 implementing a general frame expansion.

where we have performed a singular value decomposition (1.213) on the Hermitian matrix  $\tilde{\Phi} \tilde{\Phi}^T$  via (1.223a) as

$$\tilde{\Phi} \tilde{\Phi}^T = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} = \underbrace{\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}}_U \underbrace{\begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}}_\Sigma \underbrace{\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}}_{U^T}. \quad (10.25a)$$

Because  $\tilde{\Phi} \tilde{\Phi}^T$  is a Hermitian matrix, (1.225) holds, that is,

$$\lambda_{\min} I \leq \tilde{\Phi} \tilde{\Phi}^T \leq \lambda_{\max} I, \quad (10.25b)$$

where  $\lambda_{\min}$  and  $\lambda_{\max}$  are the smallest and largest eigenvalues of  $\tilde{\Phi} \tilde{\Phi}^T$ . Thus, with  $\lambda_{\min} = 1$ ,  $\lambda_{\max} = 3$ , we get

$$\|x\|^2 \leq \|\alpha\|^2 = \|\tilde{\Phi}^T x\|^2 \leq 3\|x\|^2. \quad (10.26)$$

Therefore, although energy is not preserved, it is bounded from below and above by the eigenvalues of  $\tilde{\Phi} \tilde{\Phi}^T$ . Depending on the range between the minimum and maximum eigenvalues, the energy can fluctuate; in general, the closer (tighter) the eigenvalues are, the better-behaved the frame is.<sup>135</sup> The set of inequalities (10.26) is similar to how Riesz bases were defined in (1.80), and a similar relation holds for  $\Phi$ ,

$$\frac{1}{\lambda_{\max}} I \leq \Phi \Phi^T \leq \frac{1}{\lambda_{\min}} I,$$

and thus

$$\frac{1}{3} \|x\|^2 \leq \|\Phi^T x\|^2 \leq \|x\|^2.$$

This frame is related to the previous one in the same way a biorthogonal basis is related to an orthonormal basis, and is called a *general frame*.

**Filter-Bank Implementation** In parallel to what we have done for (10.1), we can use this finite-dimensional frame as an expansion for sequences in  $\ell^2(\mathbb{Z})$  by slicing the input sequence into pieces of length 2 and applying the frame we just saw to

<sup>135</sup>This explains the word *tight* in tight frames (where the eigenvalues are equal).

each of these. To form a frame for  $\ell^2(\mathbb{Z})$ , we form three template frame vectors from the three  $\mathbb{R}^2$  vectors (10.18)

$$\varphi_0 = \begin{bmatrix} \vdots \\ 0 \\ \boxed{1} \\ 0 \\ 0 \\ \vdots \end{bmatrix}, \quad \varphi_1 = \begin{bmatrix} \vdots \\ 0 \\ \boxed{0} \\ 1 \\ 0 \\ \vdots \end{bmatrix}, \quad \varphi_2 = \begin{bmatrix} \vdots \\ 0 \\ \boxed{-1} \\ -1 \\ 0 \\ \vdots \end{bmatrix}. \quad (10.27)$$

To form the dual frame, we form three template vectors from the three  $\mathbb{R}^2$  dual frame vectors, (10.22), leading to the frame  $\Phi$  and dual frame  $\tilde{\Phi}$ :

$$\Phi = \{\varphi_{0,n-2k}, \varphi_{1,n-2k}, \varphi_{2,n-2k}\}_{k \in \mathbb{Z}}, \quad \tilde{\Phi} = \{\tilde{\varphi}_{0,n-2k}, \tilde{\varphi}_{1,n-2k}, \tilde{\varphi}_{2,n-2k}\}_{k \in \mathbb{Z}}.$$

Renaming the template frame sequences  $\varphi_{i,n} = g_{i,n}$  and the dual ones  $\tilde{\varphi}_{i,n} = \tilde{g}_{i,-n}$ , we again have a 3-channel filter bank with down/upsampling by 2, as in Figure 10.4.

### Choosing the Frame Expansion and Expansion Coefficients

So far, we have seen two redundant representations, a tight frame (10.3), akin to an orthonormal basis, and a general frame (10.19), akin to a biorthogonal basis. We showed properties, including robustness to noise and loss. Given a sequence  $x$ , how do we then choose an appropriate frame expansion? Moreover, as we have already mentioned, we can have infinitely many dual frames, and thus, infinitely many expansion coefficients  $\alpha'$ , which one to choose? We tackle these questions in the next section; here, we just show a simple example that indicates the trade-offs.

**Choosing the Frame Expansion** Assume we are working in  $\mathbb{R}^N$  and we are given an input sequence  $x$  consisting of a single complex sinusoidal sequence of unknown frequency  $(2\pi/N)\ell$  and a Kronecker delta sequence of unknown location  $k$ :

$$x_n = \beta_1 e^{j(2\pi/N)\ell n} + \beta_2 \delta_{n-k}.$$

As discussed in Chapter 6, we can use a length- $N$  DFT to expand  $x$ ; we know this will effectively localize the sinusoid in frequency, but will do a poor job in time, and the location of the Kronecker delta impulse will be essentially lost. We can use the dual, standard basis, with the dual effect: it will do an excellent job of localizing the Kronecker delta impulse but will fail in localizing the frequency of the sinusoid.

While we could use a wavelet representation from Chapter 9, an even more obvious option is to use both bases at the same time, effectively creating a frame with the following  $2N$  frame vectors:

$$\Phi = [\text{DFT} \quad I]_{N \times 2N}, \quad (10.28)$$

where the first  $N$  are the DFT basis vectors (2.160),

$$\varphi_i = \frac{1}{\sqrt{N}} \begin{bmatrix} W_N^0 & W_N^i & \dots & W_N^{i(N-1)} \end{bmatrix}^T,$$

while the last  $N$  are the standard basis vectors  $\delta_{n-i}$ . Using (1.136), we see that this is a tight frame since<sup>136</sup>

$$\Phi \Phi^* = \begin{bmatrix} \text{DFT} & I \end{bmatrix} \begin{bmatrix} \text{DFT}^* \\ I \end{bmatrix} = \underbrace{\text{DFT} \text{DFT}^*}_I + I = 2I. \quad (10.29)$$

**Choosing the Expansion Coefficients** As  $x$  has only two components, there exists a way to write it as

$$x = \Phi \alpha',$$

where  $\alpha' = \tilde{\Phi}^* x$  has exactly 2 nonzero coefficients<sup>137</sup>

$$\alpha' = [0 \quad \dots \quad 0 \quad \beta_1 \quad 0 \quad \dots \quad 0 \quad \beta_2 \quad 0 \quad \dots \quad 0]^T,$$

where  $\beta_1$  is at the  $\ell$ th location and  $\beta_2$  at the  $(N+k)$ th location. Such an expansion is called *sparse*, in the sense that it uses a small number of frame vectors. This is different from  $\alpha$  obtained from the canonical dual  $\tilde{\Phi} = (1/2)\Phi$ ,

$$\alpha = \frac{1}{2} \Phi^* x,$$

which has two dominant components at the same locations as  $\alpha'$ , but also many more nonzero components. We will see later that, while  $\alpha'$  has fewer nonzero coefficients,  $\alpha$  has a smaller  $\ell^2$  norm (see Solved Exercise 10.1).<sup>138</sup> This is an important message; while in bases, the expansion coefficients are always unique, in frames they are not, and minimizing different norms will lead to different expansions.

## Chapter Outline

This chapter is somewhat unusual in its scope. While most of the chapters in Part II deal either with Fourier- or wavelet-like expansions, this chapter deals with both. However, there is one important distinction: these expansions are all overcomplete, or, redundant. Thus, our decision to keep them all in one chapter.

Unlike for bases, where we have discussed standard finite-dimensional expansions such as the DFT, we have not done so with frames until now, and thus, Section 10.2 investigates finite-dimensional frames. We then resume the structure we have been following starting with Chapter 7, that is, we discuss the signal-processing vehicle for implementing frame expansions—oversampled filter banks. We follow with local Fourier frames in Section 10.4 and wavelet frames in Section 10.5. Section 10.6 concludes with computational aspects.

The sections that follow can also be seen as redundant counterparts of previous chapters. For example, Section 10.2 on finite-dimensional frames, has its basis counterpart in Chapters 1 and 2, where we discussed finite-dimensional bases in general (Chapter 1) as well as some specific ones, such as the DFT (Chapter 2).

<sup>136</sup>Note that now we use the Hermitian transpose of  $\Phi$  as it contains complex entries.

<sup>137</sup>Although it might not be obvious how to calculate that expansion.

<sup>138</sup>In fact,  $\alpha'$  can be chosen to minimize the  $\ell^1$  norm, while  $\alpha$  minimizes the  $\ell^2$  norm.

Section 10.3 on oversampled filter banks has its basis (critically-sampled filter bank) counterpart in Chapter 7 (two-channel critically-sampled filter banks), Chapter 8 ( $N$ -channel critically-sampled filter banks) and Chapter 9 (tree-structured critically-sampled filter banks). Section 10.4 on local Fourier frames has its basis counterpart in Chapter 8 (local Fourier bases on sequences), while Section 10.5 on wavelet frames has its basis counterpart in Chapter 9 (wavelet bases on sequences). Thus, this chapter also plays a unifying role in summarizing concepts on expansions of sequences. The two chapters that follow this one will deal with functions instead of sequences.

## 10.2 Finite-Dimensional Frames

We have just seen examples showing that finite-dimensional overcomplete sets of vectors have properties similar to orthonormal and/or biorthogonal bases. We will now look into general properties of such finite-dimensional frames in  $\mathbb{C}^N$ , with the understanding that  $\mathbb{R}^N$  is just a special case. We start with tight frames and follow with general frames. Since the representation in a frame is in general nonunique, we discuss how to compute expansion coefficients, and point out that, depending on which norm is minimized, different solutions are obtained.

Finite-dimensional frames are represented via rectangular matrices; thus, all the material in this section is basic linear algebra. We use this simplicity to develop the geometric intuition to be carried to infinite-dimensional frames that follow.

### 10.2.1 Tight Frames for $\mathbb{C}^N$

We work in a finite-dimensional space  $\mathbb{C}^N$ , where, a set of vectors  $\Phi = \{\varphi_i\}_{i=0}^{M-1}$ ,  $M > N$ , is a frame represented (similarly to (10.3) and (10.19)) by the frame matrix  $\Phi$  as

$$\Phi = [\varphi_0 \ \varphi_1 \ \dots \ \varphi_{M-1}]_{N \times M}. \quad (10.30)$$

Assume that the  $\text{rank}(\Phi) = N$ , that is, the column range of  $\Phi$  is  $\mathbb{C}^N$ . Thus, any  $x \in \mathbb{C}^N$  can be written as a nonunique linear combination of  $\varphi_i$ 's.

We impose a further constraint and start with frames that satisfy a Parseval-like equality:

**DEFINITION 10.1 (TIGHT FRAME)** A family  $\Phi = \{\varphi_i\}_{i=0}^{M-1}$  in  $\mathbb{C}^N$  is called a *tight frame*, or,  $\lambda$ -*tight frame* when there exists a constant  $0 < \lambda < \infty$  called the *frame bound*, such that for all  $x \in \mathbb{C}^N$ ,

$$\lambda \|x\|^2 = \sum_{i=0}^{M-1} |\langle x, \varphi_i \rangle|^2 = \|\Phi^* x\|^2. \quad (10.31)$$

**Expansion**

Equation (10.31) has a number of consequences. First, it means that

$$\Phi \Phi^* = \lambda I. \quad (10.32)$$

Thus,  $\Phi^*$  is a right inverse of  $\Phi$ . Calling  $\tilde{\Phi}$  the dual frame as before, we see that

$$\tilde{\Phi} = \frac{1}{\lambda} \Phi. \quad (10.33)$$

Then:

$$x = \Phi \alpha, \quad \alpha = \frac{1}{\lambda} \Phi^* x, \quad (10.34a)$$

$$x = \frac{1}{\lambda} \sum_{i=0}^{M-1} \langle x, \varphi_i \rangle \varphi_i. \quad (10.34b)$$

This looks very similar to an orthonormal basis expansion, except for the scaling factor  $(1/\lambda)$  and the fact that  $\Phi = \{\varphi_i\}_{i=0}^{M-1}$  cannot be a basis since  $\varphi_i$  are not linearly independent. We can pull the factor  $(1/\lambda)$  into the sum and renormalize the frame vectors as  $\varphi'_i = (1/\sqrt{\lambda})\varphi_i$  leading to the expression that formally looks identical to that of an orthonormal basis expansion:

$$x = \sum_{i=0}^{M-1} \langle x, \varphi'_i \rangle \varphi'_i. \quad (10.35)$$

We have already seen an example of such a renormalization in (10.1) and (10.13). A frame normalized so that  $\lambda = 1$  is called a *Parseval* tight frame or a 1-tight frame.

The expression for  $x$  is what is typically called a *reconstruction* or a *representation* of a sequence, or, in filter banks, *synthesis*, while the expression for the expansion coefficients  $\alpha$  is a *decomposition*, or, *analysis* in filter banks.

In the discussion above, we said nothing about the norms of the individual frame vectors. Since the analysis computes inner products  $\alpha_i = \langle x, \varphi_i \rangle$ , it often makes sense for all  $\varphi_i$  to have the same norm, leading to an *equal-norm* frame (which may not be tight). When we combine equal norm with tightness, we get a frame that acts every bit like an orthonormal basis, except for the redundancy. That is, all inner products  $\alpha_i = \langle x, \varphi_i \rangle$  are projections of  $x$  onto vectors of the same norm, allowing us to compare coefficients  $\alpha_i$  to each other. Moreover, because the frame is tight, the right inverse is simply its adjoint (within scaling). In finite dimensions, tightness corresponds to the rows of  $\Phi$  being orthogonal. Because of this, it is hard in general to obtain an equal-norm frame starting from the tight one.

**Geometry of the Expansion** Let us explore the geometry of tight frames. With the frame matrix  $\Phi$  as in (10.30), and as we did in (10.8), we introduce  $\Phi^\perp$ ,

$$\Phi^\perp = [\varphi_0^\perp \quad \varphi_1^\perp \quad \dots \quad \varphi_{M-1}^\perp]_{(M-N) \times M} \quad (10.36)$$

## 10.2. Finite-Dimensional Frames

727

as one possible orthogonal complement of  $\Phi$  in  $\mathbb{C}^M$ :

$$\begin{bmatrix} \Phi \\ \Phi^\perp \end{bmatrix} = \begin{bmatrix} \varphi_0 & \varphi_1 & \cdots & \varphi_{M-1} \\ \varphi_0^\perp & \varphi_1^\perp & \cdots & \varphi_{M-1}^\perp \end{bmatrix}_{M \times M} \quad (10.37)$$

that is, the rows of  $\Phi^\perp$  are chosen to be orthogonal to the rows of  $\Phi$ , orthogonal to each other and of norm 1, or,

$$\begin{aligned} \begin{bmatrix} \Phi \\ \Phi^\perp \end{bmatrix} \begin{bmatrix} \Phi^* & (\Phi^\perp)^* \end{bmatrix} &= \begin{bmatrix} \Phi\Phi^* & \Phi(\Phi^\perp)^* \\ \Phi^\perp\Phi^* & \Phi^\perp(\Phi^\perp)^* \end{bmatrix} \\ &= \begin{bmatrix} I_{N \times N} & 0 \\ 0 & I_{(M-N) \times (M-N)} \end{bmatrix} = I_{M \times M}. \end{aligned} \quad (10.38)$$

Note that each vector  $\varphi_i$  is in  $\mathbb{C}^N$ , while each vector  $\varphi_i^\perp$  is in  $\mathbb{C}^{M-N}$ . We can rewrite (10.37) as

$$S = \text{span}(\Phi^T) \subset \mathbb{C}^M, \quad (10.39a)$$

$$S^\perp = \text{span}((\Phi^\perp)^T) \subset \mathbb{C}^M, \quad (10.39b)$$

$$\mathbb{C}^M = S \oplus S^\perp, \quad (10.39c)$$

and, because of (10.38),

$$\Phi\Phi^* = I_{N \times N}, \quad (10.40a)$$

$$\Phi(\Phi^\perp)^* = 0_{N \times (M-N)}, \quad (10.40b)$$

$$\Phi^\perp(\Phi^\perp)^* = I_{(M-N) \times (M-N)}. \quad (10.40c)$$

A vector of expansion coefficients  $\alpha' \in \mathbb{C}^M$  can be written as

$$\alpha' = \alpha + \alpha^\perp \quad (10.41a)$$

with  $\alpha \in S$  and  $\alpha^\perp \in S^\perp$ , and thus

$$\langle \alpha, \alpha^\perp \rangle = 0, \quad (10.41b)$$

as we have already seen in the simple example in the previous section, (10.11b).

**Relation to Orthonormal Bases**

We now look into connections between tight frames and orthonormal bases.

**THEOREM 10.2** A 1-tight frame with unit-norm vectors is an orthonormal basis.

*Proof.* For a tight frame expansion with  $\lambda = 1$ ,

$$T = \Phi\Phi^* = I,$$

and thus, all the eigenvalues of  $T$  are equal to 1. Using one of the useful frame relations we introduce later in (10.64a),

$$N \stackrel{(a)}{=} \sum_{j=0}^{N-1} \lambda_j \stackrel{(b)}{=} \sum_{i=0}^{M-1} \|\varphi_i\|^2 \stackrel{(c)}{=} M,$$

where (a) follows from all eigenvalues being equal to 1; (b) from (10.64a); and (c) from all frame vectors being of unit norm. We get that  $M = N$ , and thus, our frame is an orthonormal basis.

In this result, we see once more the tantalizing connection between tight frames and orthonormal bases. In fact, even more is true: tight frames and orthonormal bases arise from the minimization of the same quantity called the *frame potential*:

$$\text{FP}(\Phi) = \sum_{i,j=0}^{M-1} |\langle \varphi_i, \varphi_j \rangle|^2. \quad (10.42)$$

In fact, minimizing the frame potential has two possible outcomes:

- (i) When  $M \leq N$ , the minimum value of the frame potential is

$$\text{FP}(\Phi) = N,$$

achieved when  $\Phi$  is an orthonormal set.

- (ii) When  $M > N$ , the minimum value of the frame potential is

$$\text{FP}(\Phi) = \frac{M^2}{N},$$

achieved when  $\Phi$  is a unit-norm tight frame.<sup>139</sup>

This tells us that unit-norm tight frames are a natural extension of orthonormal bases, that is, the theorem formalizes the intuitive notion that unit-norm tight frames are a generalization of orthonormal bases. Moreover, both orthonormal bases and unit-norm tight frames are results of the minimization of the frame potential, with different parameters (number of elements equal/larger than the dimension of the space). We give pointers to more details on this topic in *Further Reading*.

**EXAMPLE 10.1 (TIGHT FRAMES AND ORTHONORMAL BASES)** We illustrate this result with an example. Fix  $N = 2$ .

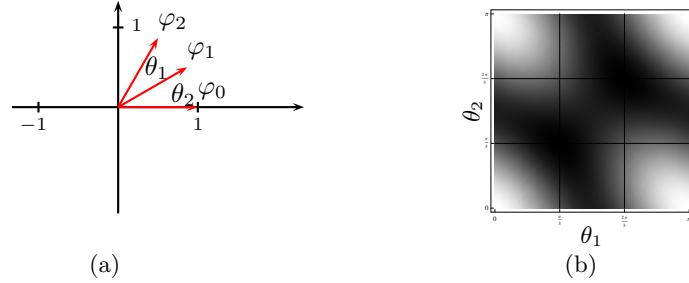
- (i) We first consider the case when  $M = N = 2$ . Then, we have two vectors only,  $\varphi_0$  and  $\varphi_1$ , both on the unit circle. According to (10.42), the frame potential is

$$\text{FP}(\{\varphi_0, \varphi_1\}) = \|\varphi_0\|^2 + \|\varphi_1\|^2 + 2|\langle \varphi_0, \varphi_1 \rangle|^2 \stackrel{(a)}{=} 2(1 + |\langle \varphi_0, \varphi_1 \rangle|^2), \quad (10.43)$$

where (a) follows from  $\varphi_0, \varphi_1$ , being of unit norm. The above expression is minimized when  $\langle \varphi_0, \varphi_1 \rangle = 0$ , that is, when  $\varphi_0$  and  $\varphi_1$  form an orthonormal basis. In that case, the minimum of the frame potential is  $FP = 2 = N$ .

<sup>139</sup>This lower bound for frames is known as *Welch bound* arising when minimizing interuser interference in a CDMA system (see *Further Reading* for pointers).





**Figure 10.5:** Minimization of the frame potential for frames with unit-norm vectors. (a) Three unit-norm vectors in  $\mathbb{R}^2$ . (b) Density plot of the frame potential as a function of angles between frame vectors. The two minima are identical and appear for  $\theta_1 = \pi/3$ ,  $\theta_2 = \pi/3$  and  $\theta_1 = 2\pi/3$ ,  $\theta_2 = 2\pi/3$ .

- (ii) We now look at  $M$  larger than  $N$ ; we choose  $M = 3$ . Let us fix  $\varphi_0 = [1 \ 0]$ ;  $\varphi_1$  is  $\theta_1$  away from  $\varphi_0$  in counterclockwise direction;  $\varphi_2$  is  $\theta_2$  away from  $\varphi_1$  in counterclockwise direction (see Figure 10.5(a)). The frame potential is now

$$FP(\{\theta_1, \theta_2\}) = \|\varphi_0\|^2 + \|\varphi_1\|^2 + \|\varphi_2\|^2 + \quad (10.44)$$

$$2(|\langle \varphi_0, \varphi_1 \rangle|^2 + |\langle \varphi_0, \varphi_2 \rangle|^2 + |\langle \varphi_1, \varphi_2 \rangle|^2) \\ = 3 + 2(\cos \theta_1 + \cos \theta_2 + \cos(\theta_1 + \theta_2)). \quad (10.45)$$

Figure 10.5(b) shows the density plot of  $FP(\{\theta_1, \theta_2\})$  for  $\theta_i \in [0, \pi]$ . From the figure, we see that there are two minima, for  $\theta_1 = \theta_2 = \pi/3$  and  $\theta_1 = \theta_2 = 2\pi/3$ , both of which lead to tight frames; the second choice is the frame we have seen in (10.13), the first choice is, in fact, identical to the second (within reflection). We thus see that the results of minimizing the frame potential in this case are tight frames, with the minimum of

$$FP(\{\frac{\pi}{3}, \frac{\pi}{3}\}) = FP(\{2\frac{\pi}{3}, 2\frac{\pi}{3}\}) = 3 + 2(\frac{1}{4} + \frac{1}{4} + \frac{1}{4}) = \frac{9}{2} = \frac{M^2}{N},$$

as per the theorem.

This simple example shows that minimizing the frame potential with different parameters leads to either orthonormal sets (orthonormal bases when  $M = N$ ) or unit-norm tight frames.

**Naimark's Theorem** Another powerful connection between orthonormal bases and tight frames is also a constructive way to obtain all tight frames. It is given by the following theorem, due to Naimark, which says that all tight frames can be obtained by projecting orthonormal bases from a larger space (of dimension  $M$ ) onto a smaller one (of dimension  $N$ ). We have seen one such example in (10.8), where the frame  $\Phi \in \mathbb{R}^2$  from (10.3) was obtained by projecting an orthonormal basis from  $\mathbb{R}^3$ .

**THEOREM 10.3** (NAIMARK [3], HAN & LARSON [67]) A frame  $\Phi \in \mathbb{C}^N$  is tight if and only if there exists an orthonormal basis  $\Psi \in \mathbb{C}^M$ ,  $M \geq N$ , such that

$$\Phi^* = \Psi[J], \quad (10.46)$$

where  $J \subset \{0, 1, \dots, M-1\}$  is the index set of the retained columns of  $\Psi$ , a process known as *seeding*.

Here, we considered only the tight-frame finite-dimensional instantiation of the theorem. For general finite-dimensional frames, a similar result holds, that is, any frame can be obtained by projecting a biorthogonal basis from a larger space. In Theorem 10.8 we formulate the statement for infinite-dimensional frames implementable by oversampled filter banks.

*Proof.* Given is a tight frame  $\Phi$ , with columns  $\varphi_i$ ,  $i = 0, 1, \dots, M-1$ , and rows  $\psi_j$ ,  $j = 0, 1, \dots, N-1$ . Because  $\Phi$  is a tight frame, it satisfies (10.32); without loss of generality, renormalize it by  $(1/\sqrt{\lambda})$  so that the frame we work with is 1-tight. This further means that

$$\langle \psi_i, \psi_j \rangle = \delta_{i-j},$$

that is,  $\{\psi_0, \psi_1, \dots, \psi_{N-1}\}$  is an orthonormal set, and, according to (10.39a), it spans the subspace  $S \subset \mathbb{C}^M$ . The whole proof in this direction follows from the geometry of tight frames we discussed earlier, by showing, as we did in (10.37), how to complete the tight frame matrix  $\Phi$  to obtain an orthonormal basis  $\Psi^*$ .

The other direction is even easier. Assume we are given a unitary  $\Psi$ . Choose any  $N$  columns of  $\Psi$  and call the resulting  $M \times N$  matrix  $\Phi^*$ . Because these columns form an orthonormal set, the rows of  $\Phi$  form an orthonormal set, that is,

$$\Phi\Phi^* = I.$$

Therefore,  $\Phi$  is a tight frame.

**Harmonic Tight Frames** We now look into an example of a well-known family of tight frames called *harmonic tight frames*, a representative of which we have already seen in (10.3). Harmonic tight frames are frame counterparts of the DFT, and are, in fact, obtained from the DFT by seeding, a process defined in Theorem 10.3.

Specifically, to obtain harmonic tight frames, we start with the DFT matrix  $\Psi = \text{DFT}_M$  given in (2.161a) and delete its last  $(M-N)$  columns, yielding:

$$\Phi = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_M & W_M^2 & \dots & W_M^{M-1} \\ 1 & W_M^2 & W_M^4 & \dots & W_M^{2(M-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_M^{(N-1)} & W_M^{(N-1) \cdot 2} & \dots & W_M^{(N-1)(M-1)} \end{bmatrix}, \quad (10.47a)$$

with the corresponding frame vectors

$$\varphi_i = \begin{bmatrix} W_M^0 & W_M^i & \dots & W_M^{i(N-1)} \end{bmatrix}^T, \quad (10.47b)$$

## 10.2. Finite-Dimensional Frames

731

for  $i = 0, 1, \dots, M-1$ , where  $W_M = e^{-j2\pi/M}$  is the principal  $M$ th root of unity (2.276). The norm of each frame vector is

$$\|\varphi_i\|^2 = \varphi_i^* \varphi_i = N,$$

the frame is  $M$ -tight

$$\Phi\Phi^* = MI,$$

and the Parseval-like equality is

$$\|\Phi^*x\|^2 = M\|x\|^2.$$

In its unit-norm version, we can compute its redundancy as  $(M/N)$ . Harmonic tight frames have a number of other interesting properties, some of which are explored in Exercise 10.3.

For example, we explore an interesting property of frames that holds for harmonic tight frames.

**DEFINITION 10.4 (FRAME MAXIMALLY ROBUST TO ERASURES)** A frame  $\Phi$  is called *maximally robust to erasures* when its every  $N \times N$  submatrix is invertible.

We have seen one example of a frame maximally robust to erasures: every  $2 \times 2$  submatrix of (10.3) is invertible. The motivation in that example was that such frames can sustain a loss of a maximum number of expansion coefficients and still afford perfect reconstruction of the original vector. In fact, harmonic tight frames in general possess such a property since every  $N \times N$  submatrix of (10.47a) is invertible (it is a Vandermonde matrix whose determinant is always nonzero, see (1.231) and Exercise 10.4).

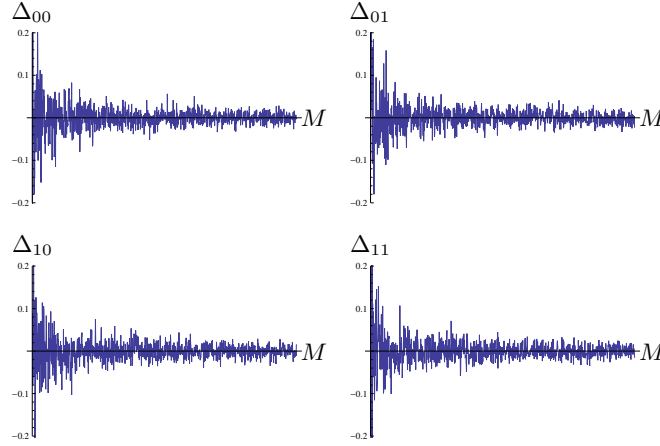
### Random Frames

While it seems that the tight frames as those we have just seen are very special, it turns out that any unit-norm frame with high redundancy will be almost tight. This is made precise by the following result:

**THEOREM 10.5 (TIGHTNESS OF RANDOM FRAMES [62])** Let  $\{\Phi_M\}_{M=N}^\infty$  be a sequence of frames in  $\mathbb{R}^N$  such that  $\Phi_M$  is generated by choosing  $M$  vectors independently with a uniform distribution on the unit sphere in  $\mathbb{R}^N$ . Then, in the mean-squared sense,

$$\frac{1}{M}\Phi\Phi^T \rightarrow \frac{1}{N}I_N \quad \text{elementwise as } M \rightarrow \infty.$$

An illustration of the theorem for  $N = 2$  is given in Figure 10.6.



**Figure 10.6:** Illustration of tightness of random frames for  $N = 2$ , and  $M = 2, 3 \dots, 1000$ . Since the convergence is elementwise, each graph plots the behavior  $\Delta_{ij} = [(1/M)\Phi\Phi^T - (1/2)I_2]_{ij}$  for  $i, j = 0, 1$ .

### 10.2.2 General Frames for $\mathbb{C}^N$

Tight frames are attractive the same way orthonormal bases are. They obey a Parseval-like energy conservation equality, and the dual frame is equal to the frame itself (the right inverse of  $\Phi$  is just its own Hermitian transpose, possibly within scaling). Tightness, however, has sometimes to be relaxed, just as orthonormality does (for example, when we wanted to design two-channel linear-phase filter banks in Chapter 7). Either a frame is given by a specific construction, and is not tight, or the constraints posed by tightness are too restrictive for a desired design.

Given a frame  $\Phi$  as in (10.30) of rank  $\text{rank}(\Phi) = N$ , we can find the *canonical* dual frame  $\tilde{\Phi}$  (formalized in Definition 10.7), also of size  $N \times M$ , made of dual frame vectors as

$$\tilde{\Phi} = (\Phi\Phi^*)^{-1}\Phi, \quad (10.48a)$$

$$= [\tilde{\varphi}_0 \ \tilde{\varphi}_1 \ \dots \ \tilde{\varphi}_{M-1}]_{N \times M}, \quad (10.48b)$$

$$\tilde{\varphi}_i = (\Phi\Phi^*)^{-1}\varphi_i. \quad (10.48c)$$

The above are all well-defined, since  $T = \Phi\Phi^*$  is of rank  $N$  and can thus be inverted. Therefore,  $\Phi$  and  $\tilde{\Phi}$  play the same roles for frames as their namesakes do for biorthogonal bases, and, using (10.48a):

$$\Phi\tilde{\Phi}^* = \Phi\Phi^*(\Phi\Phi^*)^{-1} = I_N, \quad (10.49a)$$

$$\tilde{\Phi}\Phi^* = (\Phi\Phi^*)^{-1}\Phi\Phi^* = I_N. \quad (10.49b)$$

Note that the canonical dual frame  $\tilde{\Phi}$  chosen here is a particular right inverse; we will formalize this in Definition 10.7. Note also that when  $\Phi$  is tight, with this definition of the dual, we indeed obtain  $\tilde{\Phi} = \Phi$ .

**Expansion**

We know that  $\Phi = \{\varphi_i\}_{i=0}^{M-1}$  span  $\mathbb{C}^N$ , that is, any  $x \in \mathbb{C}^N$  can be written as

$$x = \sum_{i=0}^{M-1} \alpha_i \varphi_i \quad (10.50a)$$

$$= \sum_{i=0}^{M-1} \tilde{\alpha}_i \tilde{\varphi}_i, \quad (10.50b)$$

both of which follow from (10.49) by writing

$$x \stackrel{(a)}{=} \Phi \tilde{\Phi}^* x = \Phi \alpha \quad (10.51a)$$

$$\stackrel{(b)}{=} \tilde{\Phi} \Phi^* x = \tilde{\Phi} \tilde{\alpha}, \quad (10.51b)$$

where (a) leads to (10.50a) and (b) to (10.50b), respectively. Again, these are reconstruction (representation) of a sequence, or, in filter banks, synthesis.

We have used  $\alpha_i$  and  $\tilde{\alpha}_i$  liberally, defining them implicitly. It comes as no surprise that

$$\alpha_i = \langle x, \tilde{\varphi}_i \rangle \quad \alpha = \tilde{\Phi}^* x, \quad (10.52a)$$

$$\tilde{\alpha}_i = \langle x, \varphi_i \rangle \quad \tilde{\alpha} = \Phi^* x; \quad (10.52b)$$

they are interchangeable like the expansion expressions. As before, the expression for  $\alpha$  is sometimes called decomposition (or, analysis, in filter banks).

**Geometry of the Expansion** Similarly to tight frames, let us explore the geometry of general frames. For tight frames, we dealt with  $\Phi$  and  $\Phi^\perp$  as in (10.30),(10.37); here, we must add the dual frame  $\tilde{\Phi}$  and its complement  $\tilde{\Phi}^\perp$ :

$$\begin{array}{ccc} \Phi & \tilde{\Phi} & N \times M \\ \Phi^\perp & \tilde{\Phi}^\perp & (M-N) \times M \end{array}$$

with, similarly to (10.38),

$$\Phi \tilde{\Phi}^* = I_{N \times N}, \quad (10.53a)$$

$$\Phi (\tilde{\Phi}^\perp)^* = 0_{N \times (M-N)}, \quad (10.53b)$$

$$\Phi^\perp \tilde{\Phi}^* = 0_{(M-N) \times N}, \quad (10.53c)$$

$$\Phi^\perp (\tilde{\Phi}^\perp)^* = I_{(M-N) \times (M-N)}. \quad (10.53d)$$

As in (10.39a),  $S$  is the subspace of  $\mathbb{C}^M$  spanned by the columns of  $\Phi^*$ , while  $S^\perp$  is the subspace of  $\mathbb{C}^M$  spanned by the columns of  $(\Phi^\perp)^*$ .<sup>140</sup> We will see when discussing projection operators shortly, that

$$\text{span}(\Phi^*) = \text{span}(\tilde{\Phi}^*), \quad (10.54a)$$

$$\text{span}((\Phi^\perp)^*) = \text{span}((\tilde{\Phi}^\perp)^*), \quad (10.54b)$$

<sup>140</sup>Note that the  $\text{span}(\Phi^*) = \text{span}(\Phi^T)$ .

and thus, from now on, we will use that

$$S = \text{span}(\Phi^*). \quad (10.55)$$

As before, an arbitrary vector of expansion coefficients  $\alpha' \in \mathbb{C}^M$  can be written as

$$\alpha' = \alpha + \alpha^\perp \quad (10.56a)$$

with  $\alpha \in S$  and  $\alpha^\perp \in S^\perp$ , and thus

$$\langle \alpha, \alpha^\perp \rangle = 0. \quad (10.56b)$$

**Frame Operator  $T$**  When calculating the dual frame, the so-called canonical dual of  $\Phi$ , the product  $\Phi\Phi^*$  is central; we call it

$$T_{N \times N} = \Phi\Phi^*. \quad (10.57)$$

It is a Hermitian and positive definite matrix (see (1.224)), and thus, all of its eigenvalues are real and positive. According to (1.223a),  $T$  can be diagonalized as

$$T = \Phi\Phi^* = U\Lambda U^*, \quad (10.58)$$

where  $\Lambda$  is a diagonal matrix of eigenvalues, and  $U$  a unitary matrix of eigenvectors. The largest  $\lambda_{\max}$  and smallest  $\lambda_{\min}$  eigenvalues play a special role. For tight frames,  $\lambda_{\max} = \lambda_{\min} = \lambda$ , and  $T$  is a scaled identity,  $T = \lambda I$ , as it possesses a single eigenvalue  $\lambda$  of multiplicity  $N$ .

**Energy of Expansion Coefficients** In the examples in the introduction as well as for tight frames earlier, we have seen how the energy of the expansion coefficients is conserved or bounded, (10.12), (10.26), and (10.31), respectively. We now look into it for general frames by computing the energy of the expansion coefficients  $\tilde{\alpha}$  as

$$\|\tilde{\alpha}\|^2 = \tilde{\alpha}^* \tilde{\alpha} \stackrel{(a)}{=} x^* \Phi\Phi^* x \stackrel{(b)}{=} x^* U\Lambda U^* x, \quad (10.59)$$

where (a) follows from (10.51b) and (b) from (10.57). Thus, using (10.25b),

$$\lambda_{\min} \|x\|^2 \leq \|\tilde{\alpha}\|^2 \leq \lambda_{\max} \|x\|^2. \quad (10.60a)$$

Therefore, the energy, while not preserved, is bounded from below and above by the eigenvalues of  $T$ . How close (tight) these eigenvalues are will influence the quality of the frame in question, as we will see later. The same argument above can be repeated for  $\|\alpha\|$ , leading to

$$\frac{1}{\lambda_{\max}} \|x\|^2 \leq \|\alpha\|^2 \leq \frac{1}{\lambda_{\min}} \|x\|^2. \quad (10.60b)$$

**Relation to Tight Frames** Given a general frame  $\Phi$ , we can easily transform it into a tight frame  $\Phi'$ . We do this by diagonalizing  $T$  as in (10.58). Then the tight frame is obtained as

$$\Phi' = U\Lambda^{-1/2}U^* \Phi.$$

### Frame Operators

The pair of inequalities (10.60a) leads to an alternate definition of a frame, similar in spirit to Definition 1.42 for biorthogonal bases:

**DEFINITION 10.6 (FRAME)** A family  $\Phi = \{\varphi_i\}_{i=0}^{M-1}$  in  $\mathbb{C}^N$  is called a *frame* when there exist two constants  $0 < \lambda_{\min} \leq \lambda_{\max} < \infty$ , such that for all  $x \in \mathbb{C}^N$ ,

$$\lambda_{\min} \|x\|^2 \leq \sum_{i=0}^{M-1} |\langle x, \varphi_i \rangle|^2 \leq \lambda_{\max} \|x\|^2, \quad (10.61)$$

where  $\lambda_{\min}$ ,  $\lambda_{\max}$  are called *lower* and *upper frame bounds*.

Because of (10.60a), the frame bounds are clearly the eigenvalues of  $T$  as we have seen previously. From the definition, we can also understand the meaning of  $\lambda$  we have seen in (10.31); tight frames are obtained when the two frame bounds are equal, that is, when  $\lambda_{\max} = \lambda_{\min} = \lambda$ .

The operators we have seen so far are sometimes called: *analysis frame operator*  $\tilde{\Phi}^*$ , *synthesis frame operator*  $\Phi$ , and *frame operator*  $T = \Phi\Phi^*$ . The analysis frame operator is one of many, as there exist infinitely many dual frames for a given frame  $\Phi$ . In our finite-dimensional setting, the analysis frame operator maps an input  $x \in \mathbb{C}^N$  onto a subspace of  $\mathbb{C}^M$ , namely  $\alpha = \tilde{\Phi}^*x$  belongs to the subspace  $S$  spanned by the columns of  $\Phi^*$  as we have seen in (10.55).<sup>141</sup> These, together with other frame operators introduced shortly, are summarized in Table 10.1.

Given  $x \in \mathbb{C}^N$ , the frame operator  $T = \Phi\Phi^*$  is a linear operator from  $\mathbb{C}^N$  to  $\mathbb{C}^N$ , guaranteed to be of full rank  $N$  since (10.61) ensures that  $\lambda_{\min} > 0$ . Also,

$$\lambda_{\min} I \leq T = \Phi\Phi^* \leq \lambda_{\max} I. \quad (10.62)$$

On the other hand, given  $x \in \mathbb{C}^M$ , the operator  $\Phi^*\Phi$ , which we have seen before as a projection operator (10.10) in our simple example, maps the input onto

<sup>141</sup>Remember that  $S$  is spanned by either  $\Phi^*$  or  $\tilde{\Phi}^*$ .

a subspace  $S$  of  $\mathbb{C}^M$ . We called that operator a *Gram* operator in (1.112),  $G = \Phi^* \Phi$ :

$$\begin{aligned}
 G &= \begin{bmatrix} \varphi_0^* \\ \varphi_1^* \\ \vdots \\ \varphi_{M-1}^* \end{bmatrix} \begin{bmatrix} \varphi_0 & \varphi_1 & \cdots & \varphi_{M-1} \end{bmatrix} \\
 &= \begin{bmatrix} \langle \varphi_0, \varphi_0 \rangle & \langle \varphi_0, \varphi_1 \rangle & \cdots & \langle \varphi_0, \varphi_{M-1} \rangle \\ \langle \varphi_1, \varphi_0 \rangle & \langle \varphi_1, \varphi_1 \rangle & \cdots & \langle \varphi_1, \varphi_{M-1} \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle \varphi_{M-1}, \varphi_0 \rangle & \langle \varphi_{M-1}, \varphi_1 \rangle & \cdots & \langle \varphi_{M-1}, \varphi_{M-1} \rangle \end{bmatrix}_{M \times M} \\
 &= \begin{bmatrix} \|\varphi_0\|^2 & \langle \varphi_0, \varphi_1 \rangle & \cdots & \langle \varphi_0, \varphi_{M-1} \rangle \\ \langle \varphi_0, \varphi_1 \rangle^* & \|\varphi_1\|^2 & \cdots & \langle \varphi_1, \varphi_{M-1} \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle \varphi_0, \varphi_{M-1} \rangle^* & \langle \varphi_1, \varphi_{M-1} \rangle^* & \cdots & \|\varphi_{M-1}\|^2 \end{bmatrix} = G^*. \quad (10.63)
 \end{aligned}$$

This matrix  $G$  contains correlations between different frame vectors, and, while of size  $M \times M$ , it is of rank  $N$  only.

The frame operator  $T = \Phi \Phi^*$  and the Gram operator  $G = \Phi^* \Phi$  have the same nonzero eigenvalues (see Section 1.B.2) and thus the same trace. This fact can be used to show that the sum of eigenvalues of  $T$  is equal to the sum of the norms of the frame vectors. We state this and three further useful frame facts here; their proofs are left for Exercise 10.5:

$$\sum_{j=0}^{N-1} \lambda_j = \sum_{i=0}^{M-1} \|\varphi_i\|^2, \quad (10.64a)$$

$$Tx = \sum_{i=0}^{M-1} \langle x, \varphi_i \rangle \varphi_i, \quad (10.64b)$$

$$\langle x, Tx \rangle = \sum_{i=0}^{M-1} |\langle x, \varphi_i \rangle|^2, \quad (10.64c)$$

$$\sum_{i=0}^{M-1} \langle \varphi_i, T\varphi_i \rangle = \sum_{i,j=0}^{M-1} |\langle \varphi_i, \varphi_j \rangle|^2. \quad (10.64d)$$

**Dual Frame Operators** We have already discussed the dual frame operator in (10.48a); we now formalize it a bit more.

**DEFINITION 10.7 (CANONICAL DUAL FRAME)** Given a frame satisfying (10.61), its canonical dual frame  $\tilde{\Phi}$  and dual frame vectors are:

$$\tilde{\Phi} = (\Phi \Phi^*)^{-1} \Phi = T^{-1} \Phi, \quad (10.65a)$$

$$\tilde{\varphi}_i = (\Phi \Phi^*)^{-1} \varphi_i = T^{-1} \varphi_i. \quad (10.65b)$$



From (10.60b), we can say that the dual frame  $\tilde{\Phi}$  is a frame with frame bounds  $(1/\lambda_{\max})$  and  $(1/\lambda_{\min})$ . We also see that

$$\tilde{T} = \tilde{\Phi}\tilde{\Phi}^* = T^{-1} \underbrace{\Phi\Phi^*}_T \underbrace{(T^{-1})^*}_{T^{-1}} = T^{-1}. \quad (10.66)$$

What is particular about this canonical dual frame is that among all right inverses of  $\Phi$ ,  $\tilde{\Phi}$  leads to the smallest expansion coefficients  $\alpha$  in Euclidean norm, as shown in Solved Exercise 10.2. We will also see later in this section, that expansion coefficients  $\alpha'$  other than the coefficients  $\alpha$  obtained from the canonical dual might be more appropriate when minimizing other norms (such as  $\ell^1$ - or  $\ell^\infty$  norms).

From (10.65b), we see that to compute those dual frame vectors, we need to invert  $T$ . While in finite dimensions, and for reasonable  $M$  and  $N$ , this is not a problem, it becomes an issue as  $M$  and  $N$  grow. In that case, the inverse can be computed via a series

$$T^{-1} = \frac{2}{\lambda_{\min} + \lambda_{\max}} \sum_{k=0}^{\infty} \left( I - \frac{2}{\lambda_{\min} + \lambda_{\max}} T \right)^k, \quad (10.67)$$

which converges faster when the frame bounds  $\lambda_{\min}$  and  $\lambda_{\max}$  are close, that is, when the frame is close to being tight. Solved Exercise 10.3 sketches a proof of (10.67), and Solved Exercise 10.4 illustrates it with examples.

**Projection Operators** We have seen various versions of frame operators, mapping  $\mathbb{C}^N$  to  $\mathbb{C}^N$ , as well as the Gram operator that maps  $\mathbb{C}^M$  to  $\mathbb{C}^M$ . We now look at two other operators,  $P = \tilde{\Phi}^*\Phi$  and  $\tilde{P} = \Phi^*\tilde{\Phi}$ . In fact, these are the same, as

$$P = \tilde{\Phi}^*\Phi = ((\Phi\Phi^*)^{-1}\Phi)^*\Phi = \Phi^*(\Phi\Phi^*)^{-1}\Phi = \Phi^*\tilde{\Phi} = \tilde{P}. \quad (10.68)$$

Therefore,  $P$  maps  $\mathbb{C}^M$  to a subspace of  $\mathbb{C}^M$ ,  $S$ , and is an orthogonal projection operator, as it is idempotent and self-adjoint (Definition 1.27):

$$P^2 = (\tilde{\Phi}^*\Phi) \underbrace{(\tilde{\Phi}^*\Phi)}_I \stackrel{(a)}{=} \tilde{\Phi}^*\Phi = P,$$

$$P^* = (\tilde{\Phi}^*\Phi)^* = \Phi^*\tilde{\Phi} \stackrel{(b)}{=} \Phi^*(T^{-1}\Phi) \stackrel{(c)}{=} (T^{-1}\Phi)^*\Phi = \tilde{\Phi}^*\Phi = P,$$

where (a) follows from (10.49a); (b) from (10.65a); and (c) from  $T$  being Hermitian and thus self-adjoint. This projection operator projects the input onto the column space of  $\tilde{\Phi}^*$ , or, since  $P$  and  $\tilde{P}$  are the same, onto the column space of  $\Phi^*$ . Table 10.1 summarizes various operators we have seen until now, Table 10.2 does so for frame expansions, Table 10.3 summarizes various classes of frames and their properties, while Figure 10.7 does so pictorially.

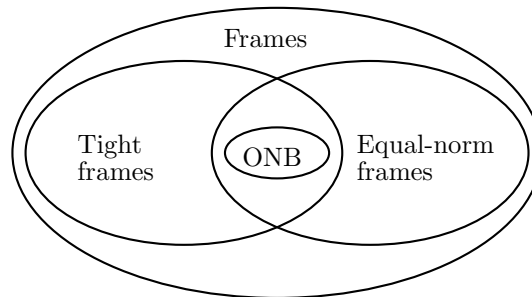
### 10.2.3 Choosing the Expansion Coefficients

Given a frame  $\Phi$  and a vector  $x$ , we have seen in (10.52a) that the expansion coefficients are given by  $\alpha = \tilde{\Phi}^*x$ ; for a tight frame, this reduces to  $\alpha = \Phi^*x$ .

Operator	Symbol	Expression	Size
Synthesis frame operator	$\Phi$		$N \times M$
Dual (analysis) frame operator	$\tilde{\Phi}$	$(\Phi\Phi^*)^{-1}\Phi$	$N \times M$
Frame operator	$T$	$\Phi\Phi^*$	$N \times N$
Gram operator	$G$	$\Phi^*\Phi$	$M \times M$
Projection operator	$P$	$\tilde{\Phi}^*\Phi$	$M \times M$

**Table 10.1:** Frame operators.

Expansion				
In $\Phi$	$x = \Phi\alpha$	$\alpha = \tilde{\Phi}^*x$	$\Phi\tilde{\Phi}^* = I$	$\lambda_{\min}I \leq T \leq \lambda_{\max}I$
In $\tilde{\Phi}$	$x = \tilde{\Phi}\tilde{\alpha}$	$\tilde{\alpha} = \Phi^*x$	$\tilde{\Phi}\Phi^* = I$	$(1/\lambda_{\max})I \leq T^{-1} \leq (1/\lambda_{\min})I$

**Table 10.2:** Frame expansions.**Figure 10.7:** Frames at a glance. Tight frames with  $\lambda = 1$  and unit-norm vectors lead to orthonormal bases.

For frames, because  $\Phi$  has a nontrivial null space, there exists an infinite set of possible expansion coefficients (see also Solved Exercise 10.2). That is, given a frame  $\Phi$  and its canonical dual  $\tilde{\Phi}$  from (10.65a), from (10.56a), we can write  $x$  as

$$x = \Phi\alpha' = \Phi(\alpha + \alpha^\perp), \quad (10.69)$$

where  $\alpha'$  is a possible vector of expansion coefficients from  $\mathbb{C}^M$ ,  $\alpha$  is its unique projection onto  $S$ , and  $\alpha^\perp$  is an arbitrary vector in  $S^\perp$ . Within this infinite set of possible expansion coefficients, we can choose particular solutions by imposing further constraints on  $\alpha'$ . Typically, this is done by minimizing a particular norm, some of which we discuss now.

## 10.2. Finite-Dimensional Frames

739

Frame	Constraints	Properties
General	$\{\varphi_i\}_{i=0}^{M-1}$ is a frame for $\mathbb{C}^N$	$\lambda_{\min}\ x\ ^2 \leq \sum_{i=0}^{M-1}  \langle x, \varphi_i \rangle ^2 \leq \lambda_{\max}\ x\ ^2$ $\lambda_{\min}I \leq T \leq \lambda_{\max}I$ $\text{tr}(T) = \sum_{j=0}^{N-1} \lambda_j = \text{tr}(G) = \sum_{i=0}^{M-1} \ \varphi_i\ ^2$
Equal-norm	$\ \varphi_i\  = \ \varphi_j\  = \varphi$ for all $i$ and $j$	$\lambda_{\min}\ x\ ^2 \leq \sum_{i=0}^{M-1}  \langle x, \varphi_i \rangle ^2 \leq \lambda_{\max}\ x\ ^2$ $\lambda_{\min}I \leq T \leq \lambda_{\max}I$ $\text{tr}(T) = \sum_{j=0}^{N-1} \lambda_j = \text{tr}(G) = \sum_{i=0}^{M-1} \ \varphi_i\ ^2 = M\varphi^2$
Tight	$\lambda_{\min} = \lambda_{\max} = \lambda$	$\sum_{i=0}^{M-1}  \langle x, \varphi_i \rangle ^2 = \lambda\ x\ ^2$ $T = \lambda I$ $\text{tr}(T) = \sum_{j=0}^{N-1} \lambda_j = N\lambda = \text{tr}(G) = \sum_{i=0}^{M-1} \ \varphi_i\ ^2$
Orthonormal basis	$\lambda_{\min} = \lambda_{\max} = 1$ $\ \varphi_i\  = 1$ for all $i$	$\sum_{i=0}^{M-1}  \langle x, \varphi_i \rangle ^2 = \ x\ ^2$ $T = I$ $\text{tr}(T) = \sum_{j=0}^{N-1} \lambda_j = N = \text{tr}(G) = \sum_{i=0}^{M-1} \ \varphi_i\ ^2 = M$ $N = M$

**Table 10.3:** Summary of properties for various classes of frames.

**Minimum  $\ell^2$ -Norm Solution** Among all expansion vectors  $\alpha'$  such that  $\Phi\alpha' = x$ , the solution with the smallest  $\ell^2$  norm is

$$\min \|\alpha'\|^2 = \|\alpha\|^2, \quad (10.70)$$

where  $\alpha = \tilde{\Phi}^*x$  is the expansion computed with respect to the canonical dual frame. The proof of this fact is along the lines of what we have seen in the introduction for the frame (10.1), since, from (10.69),

$$\|\alpha'\|^2 = \|\alpha\|^2 + \|\alpha^\perp\|^2 + 2\Re\langle \alpha, \alpha^\perp \rangle \stackrel{(a)}{=} \|\alpha\|^2 + \|\alpha^\perp\|^2,$$

where (a) follows from (10.56b). The minimum is achieved for  $\alpha^\perp = 0$ .

Since  $\Phi$  contains sets of  $N$  linearly independent vectors (often a very large number of such sets), we can write  $x$  as a linear combination of  $N$  vectors from one such set, that is,  $\alpha'$  will contain exactly  $N$  nonzero coefficients and will be sparse<sup>142</sup>. On the other hand, the minimum  $\ell^2$ -norm expansion coefficients  $\alpha$ , using the canonical dual, will typically contain  $M$  nonzero coefficients. We illustrate this in the following example:

**EXAMPLE 10.2 (NONUNIQUENESS OF THE DUAL FRAME)** Take  $\mathbb{R}^2$  and the unit-norm tight frame covering the unit circle at angles  $(2\pi i)/M$ , for  $i = 0, 1, \dots, M-$

<sup>142</sup>Remember that by *sparse* we mean an expansion that uses only  $N$  out of the  $M$  frame vectors.

1, an example of which we have already seen in (10.3) for  $M = 3$ . For  $M = 5$ , we get

$$\Phi = \begin{bmatrix} 1 & (-1 + \sqrt{5}) & -\frac{1}{4}(1 + \sqrt{5}) & -\frac{1}{4}(1 + \sqrt{5}) & \frac{1}{4}(-1 + \sqrt{5}) \\ 0 & \frac{1}{2\sqrt{2}}(\sqrt{5} + \sqrt{5}) & \frac{1}{2\sqrt{2}}(\sqrt{5} - \sqrt{5}) & -\frac{1}{2\sqrt{2}}(\sqrt{5} - \sqrt{5}) & -\frac{1}{2\sqrt{2}}(\sqrt{5} + \sqrt{5}) \end{bmatrix}.$$

The dual frame is just a scaled version of the frame itself,

$$\tilde{\Phi} = \frac{2}{5}\Phi.$$

For an arbitrary  $x$ ,  $\alpha = \tilde{\Phi}^*x$  will typically have 5 nonzero coefficients, but no fewer than 4 (when  $x$  is orthogonal to one of the  $\varphi$ 's). On the other hand, every set  $\{\varphi_i, \varphi_j\}$ ,  $i \neq j$ , is a biorthogonal basis for  $\mathbb{R}^2$ , meaning we can achieve an expansion with only 2 nonzero coefficients. Specifically, choose a biorthogonal basis  $\Psi = \{\varphi_i, \varphi_j\}$ , calculate its dual basis  $\tilde{\Psi} = \{\tilde{\varphi}_i, \tilde{\varphi}_j\}$ , and choose  $\alpha'$  as

$$\alpha'_k = \begin{cases} \langle x, \tilde{\varphi}_k \rangle, & k = i \text{ or } k = j; \\ 0, & \text{otherwise.} \end{cases}$$

We have  $\binom{5}{2} = 10$  possible bases; which ones are the best? As usual, those closer to an orthonormal basis will be better, because they are better conditioned. Thus, we should look for pairs  $\{\varphi_i, \varphi_j\}$  that have an inner product  $|\langle \varphi_i, \varphi_j \rangle|$  that is as small as possible. To do this, calculate the Gram operator (10.63) and take the absolute values of its entries  $|\langle \varphi_i, \varphi_j \rangle|$ :

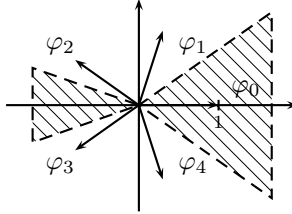
$$\frac{1}{4} \begin{bmatrix} 4 & \sqrt{5} - 1 & \sqrt{5} + 1 & \sqrt{5} + 1 & \sqrt{5} - 1 \\ \sqrt{5} - 1 & 4 & \sqrt{5} - 1 & \sqrt{5} + 1 & \sqrt{5} + 1 \\ \sqrt{5} + 1 & \sqrt{5} - 1 & 4 & \sqrt{5} - 1 & \sqrt{5} + 1 \\ \sqrt{5} + 1 & \sqrt{5} + 1 & \sqrt{5} - 1 & 4 & \sqrt{5} - 1 \\ \sqrt{5} - 1 & \sqrt{5} + 1 & \sqrt{5} + 1 & \sqrt{5} - 1 & 4 \end{bmatrix},$$

and we see, as it is obvious from the geometry of the problem, that 5 bases,  $\{\{\varphi_0, \varphi_1\}, \{\varphi_0, \varphi_4\}, \{\varphi_1, \varphi_3\}, \{\varphi_2, \varphi_4\}, \{\varphi_3, \varphi_4\}\}$ , have a minimum inner product, those with  $|\langle \varphi_i, \varphi_j \rangle| = (\sqrt{5} - 1)/4 \sim 0.31$ . Now which of these to choose? If we do not take into account  $x$ , it really does not matter. However, if we do take it into account, then it makes sense to first choose a vector  $\varphi_i$  that is most aligned with  $x$ :

$$\max_i |\langle x, \varphi_i \rangle|.$$

Assume  $\varphi_0$  is chosen, that is,  $x$  is in the shaded region in Figure 10.8. Then, either  $\varphi_1$  or  $\varphi_4$  can be used. Let us choose an  $x$  in the shaded region, say  $x = [\sqrt{3}/2 \quad 1/2]^T$ , and compute both  $\alpha = \tilde{\Phi}^*x$ , as well as  $\alpha' = \tilde{\Psi}^*x$  with the biorthogonal basis  $\Psi = \{\varphi_0, \varphi_1\}$ . Then,

$$\begin{aligned} \alpha &= [0.34641 \quad 0.297258 \quad -0.162695 \quad -0.397809 \quad -0.0831647]^T, \\ \alpha' &= [0.703566 \quad 0.525731 \quad 0 \quad 0 \quad 0]^T. \end{aligned}$$



**Figure 10.8:** Unit-norm tight frame in  $\mathbb{R}^2$ . Those  $x$  belonging to the shaded region have the maximum inner product (in magnitude) with  $\varphi_0$ . One can then choose  $\varphi_1$  or  $\varphi_4$  as the other vector in a biorthogonal expansion.

As expected,  $\alpha$  has 5 nonzero coefficients, while  $\alpha'$  has only 2. Then,

$$\|\alpha\|_2 = 0.63246 < 0.87829 = \|\alpha'\|_2, \quad (10.71)$$

as expected, as  $\alpha$  has the minimum  $\ell^2$  norm. However,

$$\|\alpha\|_1 = 1.28734 > 1.22930 = \|\alpha'\|_1, \quad (10.72)$$

and thus, the sparser expansion is worse with respect to the  $\ell^2$  norm, but is better with respect to the  $\ell^1$  norm, illustrating the wide range of possibilities for expansions in frames, as well as algorithmic issues that will be explored later.

**Minimum  $\ell^1$ -Norm Solution** Instead of the  $\ell^2$  norm, we can minimize the  $\ell^1$  norm. That is, solve

$$\min \|\alpha'\|_1 \quad \text{under the constraint} \quad \Phi\alpha' = x.$$

This can be turned into a linear program (see Section 10.6.3). Interestingly, minimizing the  $\ell^1$  norm will promote sparsity.

**EXAMPLE 10.3 (NONUNIQUENESS OF THE DUAL FRAME (CONT'D))** We now continue our previous example and calculate the expansion coefficients for the 5 biorthogonal bases  $\Psi_{01} = \{\varphi_0, \varphi_1\}$ ,  $\Psi_{04} = \{\varphi_0, \varphi_4\}$ ,  $\Psi_{13} = \{\varphi_1, \varphi_3\}$ ,  $\Psi_{24} = \{\varphi_2, \varphi_4\}$ ,  $\Psi_{34} = \{\varphi_3, \varphi_4\}$ . These, and their  $\ell^1$  norms are (we have already computed  $\alpha'_{01} = \alpha'$  above but repeat it here for completeness):

$\alpha'$	$\ \alpha'\ _1$		
$\alpha'_{01}$	0.703566	0.525731	1.22930
$\alpha'_{04}$	1.028490	-0.525731	1.55422
$\alpha'_{13}$	-0.177834	-1.138390	1.31623
$\alpha'_{24}$	-1.664120	-1.554221	3.21834
$\alpha'_{34}$	-1.028490	0.109908	1.13839

So we see that even among sparse expansions with exactly 2 nonzero coefficients there are differences. In this particular case,  $\Psi_{34}$  has the lowest  $\ell^1$  norm.

**Minimum  $\ell^0$ -Norm Solution** The  $\ell^0$  norm simply counts the number of nonzero entries in a vector:

$$\|x\|_0 = \lim_{p \rightarrow 0} \sum_{k \in \mathbb{Z}} |x_k|^p, \quad (10.73)$$

with  $0^0 = 0$ . Since a frame with  $M$  vectors in an  $N$ -dimensional space has necessarily a set of  $N$  linearly independent vectors, we can take these as a basis, compute the biorthogonal dual basis, and find an expansion  $\alpha'$  with exactly  $N$  nonzero components (as we have just done in Example 10.2). Usually, there are many such sets (see Exercise 10.4), all leading to an expansion with  $N$  nonzero coefficients. Among these multiple solutions, we may want to choose that one with the least  $\ell^2$  norm. This shows that there exists a sparse expansion, very different from the expansion that minimizes the  $\ell^2$  norm (which will typically use all  $M$  frame vectors and is thus not sparse).

**Minimum  $\ell^\infty$ -Norm Solution** Among possible expansion coefficients  $\alpha'$ , we can also choose that one that minimizes the maximum value  $|\alpha'_i|$ . That is, solve

$$\min \|\alpha'\|_\infty \quad \text{under the constraint} \quad \Phi\alpha' = x.$$

This optimization problem can be solved using TBD. While such a solution is useful when one wants to avoid large coefficients, minimizing the  $\ell^2$  norm achieves a similar goal.

**Choosing the Expansion Coefficients** In summary, we have seen that the nonuniqueness of possible frame expansion coefficients leaves us with freedom to optimize some other criteria. For example, for a sparse expansion using only a few vectors from the frame, minimizing the  $\ell^0$  norm is a possible route, although computationally difficult. Instead, minimizing the  $\ell^1$  norm achieves a similar goal (as we will see in Chapter 13), and can be done with an efficient algorithm—namely, linear programming (see Section 10.6.3). Minimizing the  $\ell^2$  norm does not lead to sparsity; instead, it promotes small coefficients, similarly to minimizing the maximum absolute value of coefficients, or the  $\ell^\infty$  norm. We illustrate this discussion with a simple example:

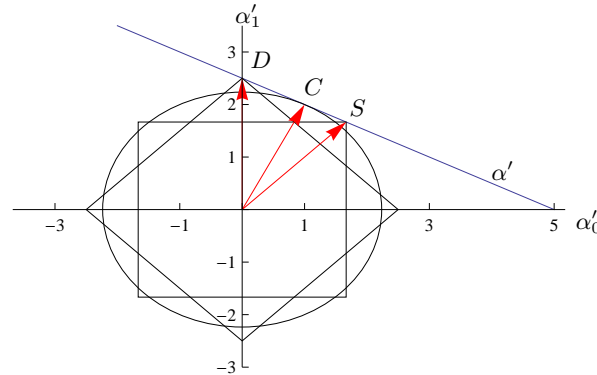
**EXAMPLE 10.4 (DIFFERENT NORMS LEAD TO DIFFERENT EXPANSIONS)** Consider the simplest example,  $N = 1$ ,  $M = 2$ . As a frame and its dual, choose

$$\Phi = \frac{1}{5} \begin{bmatrix} 1 & 2 \end{bmatrix} \quad \tilde{\Phi} = \begin{bmatrix} 1 & 2 \end{bmatrix} \quad \Phi\tilde{\Phi}^* = I.$$

Given an input  $x$ , the subspace of all expansion coefficients  $\alpha'$  that leads to  $x = \Phi\alpha'$  is described by

$$\alpha' = \alpha + \alpha^\perp = \begin{bmatrix} 1 \\ 2 \end{bmatrix} x + \begin{bmatrix} 2 \\ -1 \end{bmatrix} \gamma,$$

since the first term is colinear with  $\Phi$ , while the second is orthogonal to  $\Phi$ . In Figure 10.9 we show  $\alpha'$  for  $x = 1$ . It is a line of slope  $-1/2$  passing through



**Figure 10.9:** The space of possible expansion coefficients in the frame  $\Phi = (1/5)[1 \ 2]$ , and the subspace  $\alpha' = [1 \ 2]^T x + [2 \ -1]^T \gamma$  for  $x = 1$ . To find the points of minimum  $\ell^1$ -,  $\ell^2$ -, and  $\ell^\infty$  norms, we grow a diamond, a circle and a square, respectively, and find the intercept points with the subspace  $\alpha'$  (see also Figure 1.7 showing points with constant  $\ell^1$ -,  $\ell^2$ -, and  $\ell^\infty$  norms). These are  $D = [0 \ 5/2]$ ,  $C = [1 \ 2]$  and  $S = [5/3 \ 5/3]$ , respectively.

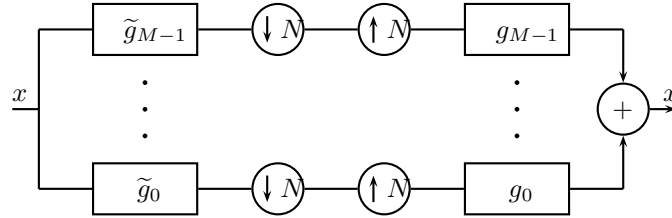
the point  $[1 \ 2]$ ,  $\alpha'_1 = -(1/2)\alpha'_0 + 5/2$ . We can choose any point on this line as a possible set  $[\alpha'_0 \ \alpha'_1]^T$  for reconstructing  $x$  with the frame  $\Phi$ . Recalling Figure 1.7 depicting points with constant  $\ell^1$ -,  $\ell^2$ -, and  $\ell^\infty$  norms, we now see what the solutions are to the minimization problem in different norms:

- (i) *Minimum  $\ell^2$ -norm solution:* The points with the same  $\ell^2$  norm form a circle. Thus, growing a circle from the origin to the intercept with  $\alpha'$  yields the point  $C = [1 \ 2]$  with the minimum  $\ell^2$  norm (see Figure 10.9). From what we know about the  $\ell^2$  norm, we could have also obtained it as the point on  $\alpha'$  closest to the origin (orthogonal projection of the origin onto the line of possible  $\alpha'$ ).
- (ii) *Minimum  $\ell^1$ -norm solution:* The points with the same  $\ell^1$  norm form a diamond. Thus, growing a diamond from the origin to the intercept with  $\alpha'$  yields the point  $D = [0 \ 5/2]$  with the minimum  $\ell^1$  norm (see Figure 10.9).
- (iii) *Minimum  $\ell^\infty$ -norm solution:* The points with the same  $\ell^\infty$  norm form a square. Thus, growing a square from the origin to the intercept with  $\alpha'$  yields the point  $S = [5/3 \ 5/3]$  with the minimum  $\ell^\infty$  norm (see Figure 10.9).

The table below numerically compares these three cases:

	$\ell^1$	$\ell^2$	$\ell^\infty$
$D$	2.50	2.50	2.50
$C$	3.00	2.24	2.00
$S$	3.33	2.36	1.67

Emphasized entries are the minimum values for each respective norm.



**Figure 10.10:** A filter-bank implementation of a frame expansion: It is an  $M$ -channel filter bank with sampling by  $N$ ,  $M > N$ .

### 10.3 Oversampled Filter Banks

This section develops necessary conditions for the design of oversampled filter banks implementing frame expansions. We consider mostly those filter banks implementing tight frames, as the general ones follow easily and can be found in the literature. As we have done for filter banks implementing basis expansions (Chapters 7-9) we also look into their polyphase representation.

From everything we have learned so far, we may expect to have an  $M$ -channel filter bank, where each channel corresponds to one of the template frame vectors (a couple of simple examples were given in Section 10.1 and illustrated in Figures 10.2 and 10.4). The infinite set of frame vectors is obtained by shifting the  $M$  template ones by integer multiples of  $N$ ,  $N < M$ ; thus the redundancy of the system. This shifting can be modeled by the samplers in the system, as we have seen previously. Not surprisingly thus, a general oversampled filter bank implementing a frame expansion is given in Figure 10.10. We now go through the salient features in some detail; however, since this material is a simple extension of what we have seen previously for bases, we will be brief.

#### 10.3.1 Tight Oversampled Filter Banks

We now follow the structure of the previous section and show the filter-bank equivalent of the expansion, expansion coefficients, geometry of the expansion, as well as look into the polyphase decomposition as a standard analysis tool, as we have done in the previous chapters.

As opposed to the previous section, we now work in an infinite-dimensional space,  $\ell^2(\mathbb{Z})$ , where formally, many things will look the same. However, we need to exercise care, and will point out specific instances when this is the case. Instead of a finite-dimensional matrix  $\Phi$  as in (10.30), we now deal with an infinite-dimensional one, and with structure: the  $M$  template frame vectors,  $\varphi_0, \varphi_1, \dots, \varphi_{M-1}$ , repeat themselves shifted in time, much the same way they do for bases. Renaming them  $g_0 = \varphi_0, g_1 = \varphi_1, \dots, g_{M-1} = \varphi_{M-1}$ , we get

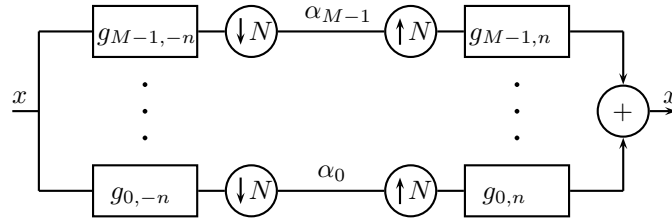
$$\Phi = \begin{bmatrix} \cdots & \boxed{g_{0,n}} & g_{1,n} & \cdots & g_{M-1,n} & g_{0,n-N} & g_{1,n-N} & \cdots & g_{M-1,n-N} & \cdots \end{bmatrix},$$

just like for critically-sampled filter banks (those with the number of channel sam-



## 10.3. Oversampled Filter Banks

745

**Figure 10.11:** A filter-bank implementation of a tight frame expansion.

ples per unit of time conserved, that is,  $M = N$ , or, those implementing basis expansions), except for the larger number of template frame vectors. We could easily implement finite-dimensional frame expansions we have seen in the last section by just limiting the number of nonzero coefficients in  $g_i$  to  $N$ , resulting in

$$\Phi = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \dots & \boxed{g_{0,0}} & \dots & g_{M-1,0} & 0 & \dots & 0 & \dots \\ \dots & g_{0,1} & \dots & g_{M-1,1} & 0 & \dots & 0 & \dots \\ \dots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \dots \\ \dots & g_{0,N-1} & \dots & g_{M-1,N-1} & 0 & \dots & 0 & \dots \\ \dots & 0 & \dots & 0 & g_{0,0} & \dots & g_{M-1,0} & \dots \\ \dots & 0 & \dots & 0 & g_{0,1} & \dots & g_{M-1,1} & \dots \\ \dots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \dots \\ \dots & 0 & \dots & 0 & g_{0,N-1} & \dots & g_{M-1,N-1} & \dots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

$$= \begin{bmatrix} \ddots & \vdots & \vdots & \ddots \\ \dots & \Phi_0 & \dots & \dots \\ \dots & & \Phi_0 & \dots \\ \ddots & \vdots & \vdots & \ddots \end{bmatrix},$$

that is, a block-diagonal matrix, with the finite-dimensional frame matrix  $\Phi_0$  of size  $N \times M$  on the diagonal. Recall that we concentrate on the tight-frame case, and therefore,  $\Phi_0 \Phi_0^* = I$ .

**Expansion** We can express the frame expansion formally in the same way as we did for finite-dimensional frames in (10.32) (again, because it is the tight-frame case)

$$\Phi \Phi^* = I, \quad (10.74)$$

except that we will always work with 1-tight frames by normalizing  $\Phi$  if necessary by  $1/\sqrt{\lambda}$ , for the filter bank to be perfect reconstruction. Writing out the expansion,

however, we see its infinite-dimensional aspect:

$$x = \sum_{i=0}^{M-1} \sum_{k \in \mathbb{Z}} \langle x, g_{i,n-Nk} \rangle g_{i,n-Nk}. \quad (10.75)$$

The process of computing the expansion coefficients is implemented via an analysis filter bank, filtering by individual filters  $g_{i,n}$ ,  $i = 0, 1, \dots, M-1$ , and downsampling by  $N$ , as on the left side of Figure 10.11:

$$\alpha = \Phi^* x \quad \alpha_{i,k} = \langle x, g_{i,n-Nk} \rangle, \quad (10.76)$$

while the process of reconstructing  $x$  is implemented via a synthesis filter bank, upsampling by  $N$  and filtering by individual filters  $g_{i,n}$ ,  $i = 0, 1, \dots, M-1$ , as on the right side of Figure 10.11:

$$x = \Phi \alpha \quad x = \sum_{i=0}^{M-1} \sum_{k \in \mathbb{Z}} \alpha_{i,k} g_{i,n-Nk}. \quad (10.77)$$

In all of the above,  $\Phi$  is an infinite matrix,  $\alpha$  and  $x$  are infinite vectors.

One can, of course, use the Fourier-domain or  $z$ -transform-domain expressions, as before. Since they are identical (except for the number of filters), we just give one as an example. For example, in  $z$ -transform-domain, we can find the expression of the effect of one single branch as

$$G_i(z) \frac{1}{N} \sum_{k=0}^{N-1} G_i(W_N^{-k} z^{-1}) X(W_N^k z).$$

Summing these over all branches,  $i = 0, 1, \dots, M-1$ , we get

$$\begin{aligned} X(z) &= \sum_{i=0}^{M-1} G_i(z) \frac{1}{N} \sum_{k=0}^{N-1} G_i(W_N^{-k} z^{-1}) X(W_N^k z) \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \left( \sum_{i=0}^{M-1} G_i(z) G_i(W_N^{-k} z^{-1}) \right) X(W_N^k z). \end{aligned}$$

Therefore, for perfect reconstruction, the term with  $X(z)$  must equal  $N$ , while all the others (aliasing terms) must cancel, that is:

$$\begin{aligned} \sum_{i=0}^{M-1} G_i(z) G_i(z^{-1}) &= N, \\ \sum_{i=0}^{M-1} G_i(z) G_i(W_N^{-k} z^{-1}) &= 0, \quad k = 1, 2, \dots, M-1. \end{aligned}$$

For example, for  $N = 2$  and  $M = 3$ , we get that:

$$\begin{aligned} G_0(z) G_0(z^{-1}) + G_1(z) G_1(z^{-1}) + G_2(z) G_2(z^{-1}) &= 2, \\ G_0(z) G_0(-z^{-1}) + G_1(z) G_1(-z^{-1}) + G_2(z) G_2(-z^{-1}) &= 0. \end{aligned}$$

Compare this to its counterpart expression in two-channel filter banks in (7.28).

**Geometry of the Expansion** Analogously to bases, each branch (channel) projects onto a subspace of  $\ell^2(\mathbb{Z})$  we call  $V_0$  or  $W_i$ ,  $i = 1, 2, \dots, M-1$ .<sup>143</sup> While each of these is on its own an orthogonal projection (because  $P_V$  in (7.18) is an orthogonal projection operator), they are not orthogonal to each other because of oversampling. Each of the orthogonal projection operators is given as

$$\begin{aligned} P_{V_0} &= G_0 U_N D_N G_0^T, \\ P_{W_i} &= G_i U_N D_N G_i^T, \quad i = 1, 2, \dots, M-1, \end{aligned}$$

with the range

$$\begin{aligned} V_0 &= \text{span}(\{g_{0,n-Nk}\}_{k \in \mathbb{Z}}), \\ W_i &= \text{span}(\{g_{i,n-Nk}\}_{k \in \mathbb{Z}}), \quad i = 1, 2, \dots, M-1. \end{aligned}$$

### 10.3.2 Polyphase View of Oversampled Filter Banks

To cover the polyphase view for general  $N$  and  $M$ , we cover it through an example with  $N = 2$ ,  $M = 3$ ; expressions for general  $N$  and  $M$  follow easily.

**EXAMPLE 10.5 (TIGHT OVERSAMPLED 3-CHANNEL FILTER BANKS)** For two-channel filter banks, a polyphase decomposition is achieved by simply splitting both sequences and filters into their even- and odd-indexed subsequences; this is governed by the sampling factor. In an oversampled tight filter bank with  $N = 2$  and  $M = 3$ , we still do the same; the difference is going to be in the number of filters, as before. We have already seen how to decompose an input sequence in (2.210), synthesis filters in (7.32), and analysis filters in (7.34). In our context, these polyphase decompositions are the same, except that for filters, we have more of them involved:

$$\begin{aligned} g_{i,0,n} &= g_{i,2n} & \xleftrightarrow{\text{ZT}} & G_{i,0}(z) = \sum_{n \in \mathbb{Z}} g_{i,2n} z^{-n}, \\ g_{i,1,n} &= g_{i,2n+1} & \xleftrightarrow{\text{ZT}} & G_{i,1}(z) = \sum_{n \in \mathbb{Z}} g_{i,2n+1} z^{-n}, \\ & & & G_i(z) = G_{i,0}(z^2) + z^{-1} G_{i,1}(z^2), \end{aligned}$$

for  $i = 0, 1, 2$  and synthesis filters. That is, we have 3 filters with 2 polyphase components each, leading to the following synthesis *polyphase matrix*  $\Phi_p(z)$ :

$$\Phi_p(z) = \begin{bmatrix} G_{0,0}(z) & G_{1,0}(z) & G_{2,0}(z) \\ G_{0,1}(z) & G_{1,1}(z) & G_{2,1}(z) \end{bmatrix}.$$

As expected, the polyphase matrix is no longer square; rather, it is a  $(2 \times 3)$  matrix of polynomials. Similarly, on the analysis side, since this is a filter bank implementing a tight frame with  $\tilde{\Phi} = \Phi$ , we assume the same filters as on the

<sup>143</sup>We assume here that the space  $V_0$  is lowpass in nature, while the  $W_i$  are bandpass.

synthesis side, only time reversed,

$$\begin{aligned}\tilde{g}_{i,0,n} &= \tilde{g}_{i,2n} = g_{i,-2n} & \xleftrightarrow{\text{ZT}} & \tilde{G}_{i,0}(z) = \sum_{n \in \mathbb{Z}} g_{i,-2n} z^{-n}, \\ \tilde{g}_{i,1,n} &= \tilde{g}_{i,2n-1} = g_{i,-2n+1} & \xleftrightarrow{\text{ZT}} & \tilde{G}_{i,1}(z) = \sum_{n \in \mathbb{Z}} g_{i,-2n+1} z^{-n}, \\ & & & \tilde{G}_i(z) = G_{i,0}(z^{-2}) + z G_{i,1}(z^{-2}),\end{aligned}$$

for  $i = 0, 1, 2$ . With this definition, the analysis polyphase matrix is, similarly to the one for the two-channel case:

$$\tilde{\Phi}_p(z) = \begin{bmatrix} G_{0,0}(z^{-1}) & G_{1,0}(z^{-1}) & G_{2,0}(z^{-1}) \\ G_{0,1}(z^{-1}) & G_{1,1}(z^{-1}) & G_{2,1}(z^{-1}) \end{bmatrix} = \Phi_p(z^{-1}),$$

where  $\tilde{\Phi}_p(z)$  is again a  $(2 \times 3)$  matrix of polynomials.

As before, this type of a representation allows for a very compact input-output relationship between the input (decomposed into polyphase components) and the result coming out of the synthesis filter bank:

$$X(z) = \begin{bmatrix} 1 & z^{-1} \end{bmatrix} \Phi_p(z^2) \Phi_p^*(z^{-2}) \begin{bmatrix} X_0(z^2) \\ X_1(z^2) \end{bmatrix},$$

where we have again used Hermitian transpose because we will often deal with complex-coefficient filter banks in this chapter. The above is formally the same as the expression for a critically-sampled filter bank with 2 channels; the oversampling is hidden in the dimensions of the rectangular matrices  $\Phi_p$  and  $\tilde{\Phi}_p$ . Clearly for the above to hold,  $\Phi_p(z^2) \Phi_p^*(z^{-2})$  must be an identity, analogously to orthogonal filter banks. This result for tight frames is formalized in Theorem 10.8.

The above example went through various polyphase concepts for a tight oversampled 3-channel filter bank. For general oversampled filter banks with  $N, M$ , expressions are the same as those given in (8.12c), (8.12e), except with  $M$  filters instead of  $N$ . The corresponding polyphase matrices are of sizes  $N \times M$  each.

**Frame Operators** All the frame operators we have seen so far can be expressed via filter bank ones as well.

The frame operator  $T$  for a general infinite-dimensional frame is formally defined as for the finite-dimensional one in (10.57), except that it is now infinite-dimensional itself. Its polyphase counterpart is:

$$T_p(z) = \Phi_p(z) \Phi_p^*(z^{-1}). \quad (10.80)$$

For a tight frame implemented by a tight oversampled filter bank, this has to be an identity as we have already said in the above example. In other words,  $\Phi_p$  is a rectangular paraunitary matrix. The frame operator  $T_p(z)$  is positive definite on the unit circle:

$$T_p(e^{j\omega}) = |\Phi_p(e^{j\omega})|^2 > 0. \quad (10.81)$$

## 10.3. Oversampled Filter Banks

749

The canonical dual frame operator has its polyphase counterpart in:

$$\tilde{\Phi}_p(z) = T_p(z)^{-1} \Phi_p(z). \quad (10.82)$$

Again, we can see that when the frame is tight,  $T_p(z) = I$ , then the dual polyphase matrix is the same as  $\Phi_p(z)$ .

**Polyphase Decomposition of an Oversampled Filter Bank** As before, the polyphase formulation allows us to characterize classes of solutions. The following theorem, the counterpart of Theorem 8.1 for critically-sampled filter banks, summarizes these without proof, the pointers to which are given in *Further Reading*.

**THEOREM 10.8 (OVERSAMPLED  $M$ -CHANNEL FILTER BANKS IN POLYPHASE DOMAIN)**

Given is an  $M$ -channel filter bank with sampling by  $N$  and the polyphase matrices  $\Phi_p(z)$ ,  $\tilde{\Phi}_p(z)$ . Then:

(i) *Frame expansion in polyphase domain*

A filter bank implements a general frame expansion if and only if

$$\Phi_p(z) \tilde{\Phi}_p^*(z) = I. \quad (10.83a)$$

A filter bank implements a tight frame expansion if and only if

$$T_p(z) = \Phi_p(z) \Phi_p^*(z^{-1}) = I, \quad (10.83b)$$

that is,  $\Phi_p(z)$  is paraunitary.

(ii) *Naimark's theorem in polyphase domain*

An infinite-dimensional frame implementable via an  $M$ -channel filter bank with sampling by  $N$  is a general frame if and only if there exists a biorthogonal basis implementable via an  $M$ -channel filter bank with sampling by  $M$  so that

$$\Phi_p^*(z) = \Psi_p(z)[J], \quad (10.84)$$

where  $J \subset \{0, \dots, M-1\}$  is the index set of the retained columns of  $\Psi_p(z)$ , and  $\Phi_p(z)$ ,  $\Psi_p(z)$  are the frame/basis polyphase matrices, respectively.

An infinite-dimensional frame implementable via an  $M$ -channel filter bank with sampling by  $N$  is a tight frame if and only if there exists an orthonormal basis implementable via an  $M$ -channel filter bank with sampling by  $M$  so that (10.84) holds.<sup>144</sup>

(iii) *Frame bounds*

The frame bounds of a frame implementable by a filter bank are given by:

$$\lambda_{\min} = \min_{\omega \in [-\pi, \pi)} T_p(e^{j\omega}), \quad (10.85a)$$

$$\lambda_{\max} = \max_{\omega \in [-\pi, \pi)} T_p(e^{j\omega}). \quad (10.85b)$$

The last statement on eigenvalues stems from the fact that the frame operator  $T$  and its polyphase counterpart  $T_p(e^{j\omega})$  are related via a unitary transformation. If the eigenvalues of  $T_p(e^{j\omega})$  are defined via  $T_p(e^{j\omega})v(\omega) = \lambda(\omega)v(\omega)$ , then the eigenvalues of  $T$  and  $T_p(e^{j\omega})$  are the same, leading to (10.85).

**EXAMPLE 10.6 (TIGHT OVERSAMPLED 3-CHANNEL FILTER BANKS CONT'D)** We now set  $M = 3$ ,  $N = 2$  and show how one can obtain a linear-phase tight frame with filters of length greater than 2, a solution not possible for critically-sampled filter banks with sampling by 2, as was shown in Proposition 7.12. We know that such a filter bank implementing a tight frame transform must be seeded from an orthogonal filter bank with a  $3 \times 3$  paraunitary matrix.

We use such a matrix in the example showing how to parameterize  $N$ -channel orthogonal filter banks, Example 8.2 with  $K = 2$ , that is, all polyphase components will be first-degree polynomials in  $z^{-1}$ . We form a tight frame by deleting its last column and call the resulting frame polyphase matrix  $\Phi_p^T(z)$ . Since there are 5 angles involved, the matrix is too big to explicitly state here; instead, we start imposing the linear-phase conditions to reduce the number of degrees of freedom. A simple solution with  $\theta_{00} = \pi/2$ ,  $\theta_{11} = \pi/2$ ,  $\theta_{02} = \pi/4$  and  $\theta_{10} = 3\pi/4$ , leads to the first two filters being symmetric of length 3 and the last antisymmetric of length 3. The resulting polyphase matrix is (where we have rescaled the first and third columns by  $-1$ ):

$$\Phi_p(z) = \begin{bmatrix} \frac{1}{2} \cos \theta_{01}(1 + z^{-1}) & \frac{1}{2} \sin \theta_{01}(1 + z^{-1}) & \frac{1}{2}(1 - z^{-1}) \\ \sin \theta_{01} & -\cos \theta_{01} & 0 \end{bmatrix}^T,$$

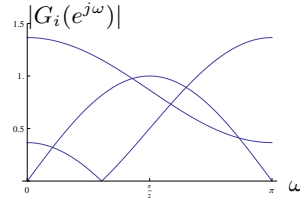
leading to the following three filters:

$$\begin{aligned} G_0(z) &= \frac{1}{2} \cos \theta_{01} + \sin \theta_{01} z^{-1} + \frac{1}{2} \cos \theta_{01} z^{-2}, \\ G_1(z) &= \frac{1}{2} \sin \theta_{01} - \cos \theta_{01} z^{-1} + \frac{1}{2} \sin \theta_{01} z^{-2}, \\ G_2(z) &= \frac{1}{2} - \frac{1}{2} z^{-2}. \end{aligned}$$

For example, with  $\theta_{01} = \pi/3$ , the three resulting filters have reasonable coverage of the frequency axis (see Figure 10.12).

## 10.4 Local Fourier Frames

Until now, the material in this chapter covered finite-dimensional frames (Section 10.2) and oversampled filter banks as a vehicle for implementing both finite-dimensional as well as certain infinite-dimensional frames (previous section). We now investigate a more specific class of frames; those obtained by modulating (shifting in frequency) a single prototype filter/frame vector, introduced in their basis form in Chapter 8. These are some of the oldest bases and frames, and some of the most widely used. The local Fourier expansions arose in response to the need to



**Figure 10.12:** Tight oversampled 3-channel filter bank with sampling by  $N = 2$  and linear-phase filters. The figure depicts the three magnitude responses.

create a *local* Fourier tool, able to achieve some localization in time, at the price of worsening the known excellent localization in frequency.

As in Chapter 8, we will consider two large classes of local Fourier frames, those obtained by complex-exponential modulation, as well as those obtained by cosine modulation of a single prototype filter/frame vector. In Chapter 8, we learned that, while there exist no good local Fourier bases (apart from those equivalent to a finite-dimensional basis), there do exist good local cosine bases. In this section, we go even farther; we show that there exist good local Fourier frames, due to the extra freedom redundancy buys us.

### 10.4.1 Complex Exponential-Modulated Local Fourier Frames

Complex-exponential modulation is used in many instances, such as the DFT basis, (8.2), (8.5), as well as the basis constructed from the ideal filters (8.6), and is at the heart of the local Fourier expansion known as Gabor transform. The term *Gabor frame* is often used to describe any frame with complex-exponential modulation and overlapping frame vectors (oversampled filter banks with filters of lengths longer than the sampling factor  $N$ ). For complex exponential-modulated bases, we defined this modulation in (8.16); for complex exponential-modulated frames, we do it now.

**Complex-Exponential Modulation** Given a prototype filter  $p = g_0$ , the rest of the filters are obtained via complex-exponential modulation:

$$\begin{aligned} g_{i,n} &= p_n e^{j(2\pi/N)in} = p_n W_M^{-in}, \\ G_i(z) &= P(W_M^i z), \\ G_i(e^{j\omega}) &= P(e^{j(\omega - (2\pi/M)i)}) = P(W_M^i e^{j\omega}), \end{aligned} \tag{10.86}$$

for  $i = 1, 2, \dots, M-1$ . A filter bank implementing such a frame expansion is often called *complex exponential-modulated oversampled filter bank*. While the prototype filter  $p = g_0$  is typically real, the rest of the bandpass filters are complex. The above is identical to the expression for bases, (8.16); the difference is in the sampling factor  $N$ , smaller here than the number of filters  $M$ .

**Overcoming the Limitations of the Balian-Low Theorem** In Chapter 8, Theorem 8.2, we saw that there does not exist a complex exponential-modulated local

Fourier basis implementable by an  $N$ -channel FIR filter bank, except for a filter bank with filters of length  $N$ . We illustrated the proof with an example for  $N = 3$  in (8.17) and demonstrated that the only solution consisted of each polyphase component being a monomial, leading to a block-based expansion.

We now investigate what happens with frames. Start with the polyphase representation (8.12d) of the prototype filter  $p = g_0$ ,

$$P(z) = P_0(z^N) + z^{-1}P_1(z^N) + \dots + z^{-(N-1)}P_{N-1}(z^N),$$

where  $P_i(z)$ ,  $i = 0, 1, \dots, N-1$  are its polyphase components. The modulated versions become

$$\begin{aligned} G_i(z) &= P(W_M^i z) \\ &= P_0(W_M^{iN} z^N) + \dots + W_M^{-(N-1)i} z^{-(N-1)} P_{N-1}(W_M^{iN} z^N), \end{aligned}$$

for  $i = 1, 2, \dots, M-1$ . On a simple example, we now show that relaxing the basis requirement allows us to implement a tight frame expansion via an oversampled filter bank with FIR filters longer than the sampling factor  $N$ .

**EXAMPLE 10.7 (OVERCOMING LIMITATIONS OF BALIAN-LOW THEOREM)** Let  $N = 2$  and  $M = 3$ . The polyphase matrix corresponding to the complex exponential-modulated filter bank is given by

$$\begin{aligned} \Phi_p(z) &= \begin{bmatrix} P_0(z) & P_0(W_3^2 z) & P_0(W_3 z) \\ P_1(z) & W_3^2 P_1(W_3^2 z) & W_3 P_1(W_3 z) \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} P_u(z) & \\ & P_\ell(z) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & W_3 \\ 0 & W_3^2 & 0 \end{bmatrix}, \quad (10.87) \end{aligned}$$

with  $P_u(z)$  and  $P_\ell(z)$  the diagonal matrices of polyphase components:

$$\begin{aligned} P_u(z) &= \text{diag}([P_0(z), P_0(W_3 z), P_0(W_3^2 z)]), \\ P_\ell(z) &= \text{diag}([P_1(z), P_1(W_3 z), P_1(W_3^2 z)]), \end{aligned}$$

and  $W_3^{-1} = W_3^2$ ,  $W_3^{-2} = W_3$ . Compare (10.87) to its basis counterpart in (8.17).

We now want to see whether it is possible for such a frame polyphase matrix



to implement a tight frame, in which case, it would have to satisfy (10.83b).

$$\begin{aligned}
& \Phi_p(z)\Phi_p^*(z^{-1}) = \\
& = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} P_u(z) & \\ & P_\ell(z) \end{bmatrix} \begin{bmatrix} I & W^{-1} \\ W & I \end{bmatrix} \begin{bmatrix} P_u(z^{-1}) & \\ & P_\ell(z^{-1}) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \\
& = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} P_u(z)P_u(z^{-1}) & W^{-1}P_u(z)P_\ell(z^{-1}) \\ W P_\ell(z)P_u(z^{-1}) & P_\ell(z)P_\ell(z^{-1}) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \\
& = \begin{bmatrix} \sum_{i=0}^2 P_0(W_3^{-i}z)P_0(W_3^i z^{-1}) & \sum_{i=0}^2 W_3^i P_0(W_3^{-i}z)P_1(W_3^i z^{-1}) \\ \sum_{i=0}^2 W_3^i P_0(W_3^{-i} z^{-1})P_1(W_3^i z) & \sum_{i=0}^2 P_1(W_3^{-i}z)P_1(W_3^i z^{-1}) \end{bmatrix} \\
& \stackrel{(a)}{=} \begin{bmatrix} \sum_{i=0}^2 P_0(W_3^i z)P_0(W_3^{-i} z^{-1}) & \sum_{i=0}^2 W_3^i P_0(W_3^i z)P_1(W_3^{-i} z^{-1}) \\ \sum_{i=0}^2 W_3^{-i} P_0(W_3^{-i} z^{-1})P_1(W_3^i z) & \sum_{i=0}^2 P_1(W_3^i z)P_1(W_3^{-i} z^{-1}) \end{bmatrix} = I,
\end{aligned}$$

where (a) follows again from  $W_3^{-1} = W_3^2$ ,  $W_3^{-2} = W_3$ ,  $W = \text{diag}([1, W_3, W_3^2])$ , and we assumed that  $p$  is real. It is clear that the set of conditions above is much less restrictive than that of every polyphase component of the prototype filter having to be a monomial (the condition that lead to the negative result in the discrete Balian-Low theorem, Theorem 8.2).

For example, we see that the conditions on each polyphase component:

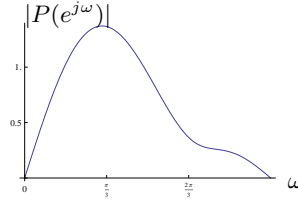
$$\begin{aligned}
\sum_{i=0}^2 P_0(W_3^i z)P_0(W_3^{-i} z^{-1}) &= 1, \\
\sum_{i=0}^2 P_1(W_3^i z)P_1(W_3^{-i} z^{-1}) &= 1,
\end{aligned}$$

are equivalent to those polyphase components being orthogonal filters as in (8.7). On the other hand, the conditions involving both polyphase components:

$$\begin{aligned}
\sum_{i=0}^2 W_3^i P_0(W_3^i z)P_1(W_3^{-i} z^{-1}) &= 0, \\
\sum_{i=0}^2 W_3^{-i} P_0(W_3^{-i} z^{-1})P_1(W_3^i z) &= 0,
\end{aligned}$$

are equivalent to  $P_0(z)$  and  $z^{-1}P_1(z)$  being orthogonal to each other as in (8.9).

For example, we know that the rows of (10.3) are orthogonal filters (since it is a tight frame and the rows are orthonormal vectors from a  $3 \times 3$  unitary



**Figure 10.13:** Magnitude response of the prototype filter  $P(z)$  of length 5.

matrix via Naimark's theorem), so we can take (with normalization)

$$P_0(z) = \frac{1}{3}(\sqrt{2} - \frac{1}{\sqrt{2}}z^{-1} - \frac{1}{\sqrt{2}}z^{-2}), \quad P_1(z) = \frac{1}{\sqrt{6}}(1 - z^{-1}).$$

We can now get the prototype filter  $P(z)$  as

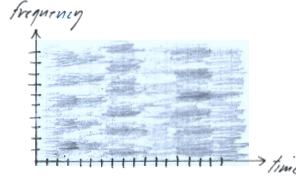
$$P(z) = P_0(z^2) + z^{-1}P_1(z^2) = \frac{1}{3\sqrt{2}}(2 + \sqrt{3}z^{-1} - z^{-2} - \sqrt{3}z^{-3} - z^{-4}),$$

a longer solutions than  $N = 2$ , with the magnitude response as in Figure 10.13. Another example, with  $N = 2$  and  $M = 4$  is left as Exercise 10.9.

**Application to Power Spectral Density Estimation** In Chapter 8, Section 8.3.2, we discussed the computation of periodograms as a widely used application of complex exponential-modulated filter banks. It is a process of estimating and computing the local power spectral density. That process has a natural filter-bank implementation described in the same section. The prototype filter  $p$  computes the windowing, and the modulation computes the DFT (see Figure 8.9 and Table 8.1). The down-sampling factor  $N$  can be smaller than  $M$ , which is when we have a frame. For example, with  $N = M/2$ , we have 50% overlap, and if  $N = 1$  (that is, no down-sampling) we are computing a sliding window DFT (with  $(M - 1)/M\%$  overlap). When both the time redundancy and the number of frequencies increases, this time-frequency frame approaches a continuous transform called the *local Fourier transform*, treated in detail in Chapter 11. A typical example for calculating the periodogram of a speech signal uses  $M = 64$ ,  $N = 32$  (or 50% overlap) and a Hamming window. No averaging of the power spectral density coefficient is used. The result is shown in Figure 10.14. This display is often called a *spectrogram* in the speech processing literature. From this figure, one clearly sees the time-frequency behavior typical of signals that have time-varying spectra.

### 10.4.2 Cosine-Modulated Local Fourier Frames

In Chapter 8, we saw that a possible escape from the restriction imposed by the discrete Balian-Low theorem was to replace complex-exponential modulation with an appropriate cosine modulation, with an added advantage that all filters are real



**Figure 10.14:** Spectrogram of a speech segment. 64 frequency bins are evaluated between 0 and 4 KHz, and a triangular window with 50% overlap is used.

if the prototype is real. While frames in general offer another such escape, cosine-modulated frames provides even more options.

**Cosine Modulation** Given a prototype filter  $p$ , one of the possible ways to use the cosine modulation is (other ways leading to different classes of cosine-modulated filter banks exist; see *Further Reading* for pointers):

$$\begin{aligned}
 g_{i,n} &= p_n \cos \left( \frac{2\pi}{2M} \left( i + \frac{1}{2} \right) n + \theta_i \right) \\
 &= p_n \frac{1}{2} \left[ e^{j\theta_i} W_{2M}^{-(i+1/2)n} + e^{-j\theta_i} W_{2M}^{(i+1/2)n} \right], \\
 G_i(z) &= \frac{1}{2} \left[ e^{j\theta_i} P(W_{2M}^{(i+1/2)} z) + e^{-j\theta_i} P(W_{2M}^{-(i+1/2)} z) \right], \\
 G_i(e^{j\omega}) &= \frac{1}{2} \left[ e^{j\theta_i} P(e^{j(\omega - (2\pi/2M)(i+1/2))}) + e^{-j\theta_i} P(e^{j(\omega + (2\pi/2M)(i+1/2))}) \right],
 \end{aligned} \tag{10.88}$$

for  $i = 0, 1, \dots, M-1$ , and  $\theta_i$  is a phase factor that gives us flexibility in designing the representation. Compare the above with (10.86) for the complex-exponential modulation; the difference is that given a real prototype filter, all the other filters are real. Compare it also with (8.27) for the cosine modulation in bases. The two expressions are identical; the difference is in the sampling factor  $N$ , smaller here than the number of filters  $M$ .

**Matrix View** We look at a particular class of cosine-modulated frames, those with filters of length  $L = 2N$ , a natural extension of the LOTs from Section 8.4.1 (see also *Further Reading*). We choose the same phase factor as in (8.29), leading to

$$g_{i,n} = p_n \cos \left( \frac{2\pi}{2M} \left( i + \frac{1}{2} \right) \left( n - \frac{M-1}{2} \right) \right), \tag{10.89}$$

for  $i = 0, 1, \dots, M-1$ ,  $n = 0, 1, \dots, 2N-1$ . We know that for a rectangular prototype window,  $p_n = 1/\sqrt{M}$ , the above filters form a tight frame since they were obtained directly by seeding the LOT with the rectangular prototype window (compare (10.89) to (8.30)). We follow the same analysis as we did in Section 8.4.1.

As in (8.34), we can express the frame matrix  $\Phi$  as

$$\Phi = \begin{bmatrix} \ddots & & & & & \\ & G_0 & & & & \\ & G_1 & G_0 & & & \\ & & G_1 & G_0 & & \\ & & & G_1 & & \\ & & & & \ddots & \end{bmatrix}, \quad (10.90)$$

except that blocks  $G_i$  that contain synthesis filters' impulse responses are now of size  $2N \times M$  (instead of  $2N \times N$ ). Given that the frame is tight (and real),

$$G_0 G_0^T + G_1 G_1^T = I, \quad (10.91a)$$

$$G_1 G_0^T = G_0 G_1^T = 0. \quad (10.91b)$$

Assume we want to impose a prototype window; then, as in (8.41), the windowed impulse responses are  $G'_0 = P_0 G_0$  and  $G'_1 = P_1 G_1$ , where  $P_0$  and  $P_1$  are the  $N \times N$  diagonal matrices with the left and right tails of the prototype window  $p$  on the diagonal, and if the prototype window is symmetric,  $P_1 = J P_0 J$ . We can thus substitute  $G'_0$  and  $G'_1$  into (10.91) to verify that the resulting frame is indeed tight

$$\begin{aligned} G'_0 G'^T_0 + G'_1 G'^T_1 &= P_0 G_0 G_0^T P_0 + P_1 G_1 G_1^T P_1 \\ &= P_0 G_0 G_0^T P_0 + J P_0 J G_1 G_1^T J P_0 J = I. \end{aligned}$$

Unlike for the LOTs,  $G_0 G_0^T$  has no special structure now; its elements are given by

$$(G_0 G_0^T)_{i,n} = t_{i,n} = \frac{1}{2M} \frac{\sin\left(\frac{\pi(i+n+1)}{2}\right)}{\sin\left(\frac{\pi(i+n+1)}{2M}\right)} + \frac{1}{2M} \frac{\sin\left(\frac{\pi(i-n)}{2}\right)}{\sin\left(\frac{\pi(i-n)}{2M}\right)},$$

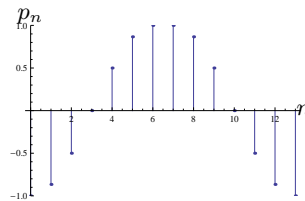
where notation  $t$  for the elements of  $G_0 G_0^T$  is evocative of the frame matrix  $T = \Phi \Phi^*$ . This leads to the following conditions on the prototype window:

$$\begin{aligned} t_{n,n} p_n^2 + (1 - t_{n,n}) p_{N-n-1}^2 &= 1, \\ p_n p_k &= p_{N-n-1} p_{N-k-1}, \end{aligned}$$

for  $n = 0, 1, \dots, N-1, k = 0, 1, \dots, N-1, k \neq n$ . We can fix one coefficient; let us choose  $p_0 = -1$ , then  $p_{N-1} = \pm 1$  and  $p_k = -p_{N-1} p_{N-k-1}$  for  $k = 1, 2, \dots, N-2$ . A possible solution for the prototype window satisfying the above conditions is

$$d_n = \begin{cases} -\cos\left(\frac{n\pi}{N-1}\right), & N = 2k+1; \\ -\cos\left(\frac{2n\pi}{N-1}\right), & N = 2k, \end{cases}$$

for  $n = 0, 1, \dots, N-1$ ; an example window design is given in Figure 10.15, with coefficients as in Table 10.4.



**Figure 10.15:** An example prototype window design for  $N = 7$ .

$p_0$	$p_1$	$p_2$	$p_3$	$p_4$	$p_5$	$p_6$
-1	$-\sqrt{3}/2$	$-1/2$	0	$1/2$	$\sqrt{3}/2$	1

**Table 10.4:** Prototype window used in Figure 10.15. The prototype window is symmetric, so only half of the coefficients are shown.

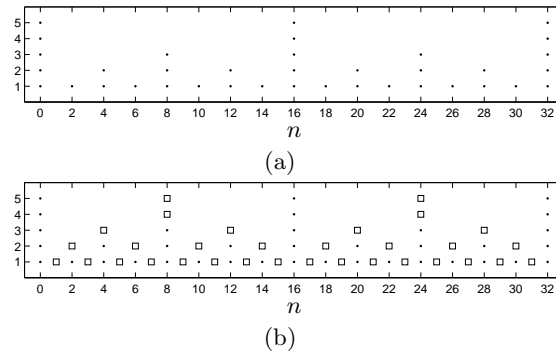
## 10.5 Wavelet Frames

We now move from the Fourier-like frames to those that are wavelet-like. We have seen examples of moving from bases to frames (DFT to harmonic tight frame, for example, see Table 10.7), and we would like to do that in the wavelet case as well. We start with the most obvious way to generate a frame from the DWT: by removing some downsamplers. Then we move on to the predecessor of wavelet frames originating in the work of Burt and Adelson on pyramid coding, and close the section with the fully-redundant frames called *shift-invariant DWT*.

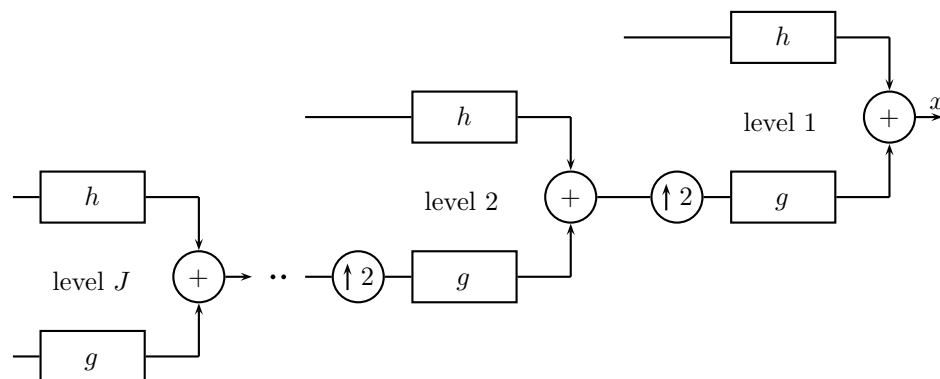
### 10.5.1 Oversampled DWT

How do we add redundancy starting from the DWT? We already mentioned that an obvious way to do that was to remove some downsamplers, thereby getting a finer time localization. Consider Figure 10.16(a), showing the sampling grid for the DWT (corresponding to the wavelet tiling from Figure 9.7(d)): at each subsequent level, only half of the points are present (half of the basis functions exist at that scale). Ideally, we would like to, for each scale, insert additional points (one point between every two). This can be achieved by having a DWT tree with the samplers removed at all free branches (see Figure 10.17). We call this scheme *oversampled DWT*, also known as the partial DWT (see *Further Reading*). The redundancy of this scheme at level  $\ell$  is  $A_j = 2$ , for a total redundancy of  $A = 2$ . The sampling grid with  $J = 4$  is depicted in Figure 10.16(b).

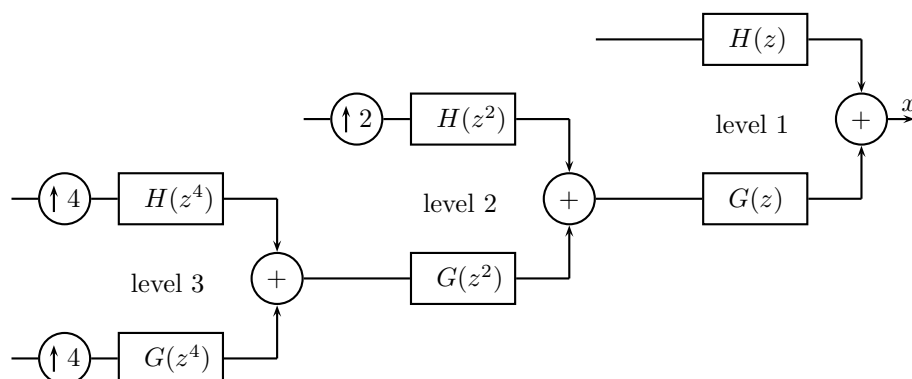
**EXAMPLE 10.8 (OVERSAMPLED DWT)** Let us now look at a simple example with  $J = 3$ . By moving upsamplers across filters, the filter bank in Figure 10.17 reduces to the one in Figure 10.18. The equivalent filters are then (we leave the



**Figure 10.16:** Sampling grids corresponding to the time-frequency tilings of (a) the DWT (points—nonredundant) and (b) the oversampled DWT (squares—redundant).



**Figure 10.17:** The synthesis part of the filter bank implementing the oversampled DWT. The samplers are omitted at all the inputs into the bank. The analysis part is analogous.



**Figure 10.18:** The synthesis part of the equivalent filter bank implementing the oversampled DWT with  $J = 3$  levels. The analysis part is analogous.

filter bank in its tree form as this is how it is actually implemented):<sup>145</sup>

$$H^{(1)}(z) = H(z), \quad (10.92a)$$

$$H^{(2)}(z) = G(z)H(z^2), \quad (10.92b)$$

$$H^{(3)}(z) = G(z)G(z^2)H(z^4), \quad (10.92c)$$

$$G^{(3)}(z) = G(z)G(z^2)G(z^4), \quad (10.92d)$$

and the frame can be expressed as

$$\Phi = \{h_{n-k}^{(1)}, h_{n-2k}^{(2)}, h_{n-4k}^{(3)}, g_{n-4k}^{(3)}\}_{k \in \mathbb{Z}}. \quad (10.93)$$

The template vector  $h$  moves by 1,  $h^{(2)}$  moves by multiples of 2, and  $h^{(3)}$  and  $g^{(3)}$  move by multiples of 4. Thus, the basic block of the infinite matrix is of size  $8 \times 16$  (the smallest period after which it starts repeating itself, redundancy of 2) and it moves by multiples of 8. However, even for filters such as Haar for which the DWT would become a block transform (the infinite matrix  $\Phi$  is block diagonal, see (9.4)), here this is not the case. Substituting Haar filters (see Table 7.8) into the expressions for  $H^{(1)}$ ,  $H^{(2)}$ ,  $H^{(3)}$  and  $G^{(3)}$  above, we get

$$\begin{aligned} H^{(1)}(z) &= \frac{1}{\sqrt{2}}(1 - z^{-1}), \\ H^{(2)}(z) &= \frac{1}{2}(1 + z^{-1} - z^{-2} - z^{-3}), \\ H^{(3)}(z) &= \frac{1}{2\sqrt{2}}(1 + z^{-1} + z^{-2} + z^{-3} - z^{-4} - z^{-5} - z^{-6} - z^{-7}), \\ G^{(3)}(z) &= \frac{1}{2\sqrt{2}}(1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + z^{-5} + z^{-6} + z^{-7}). \end{aligned}$$

Renaming the template frame vectors, we can rewrite the frame  $\Phi$  as

$$\begin{aligned} \varphi_{k,n} &= h_{n-k}^{(1)}, & k = 0, 1, \dots, 7; \\ \varphi_{8+k,n} &= h_{n-2k}^{(2)}, & k = 0, 1, 2, 3; \\ \varphi_{12+k,n} &= h_{n-4k}^{(3)}, & k = 0, 1; \\ \varphi_{14+k,n} &= g_{n-4k}^{(3)}, & k = 0, 1; \end{aligned}$$

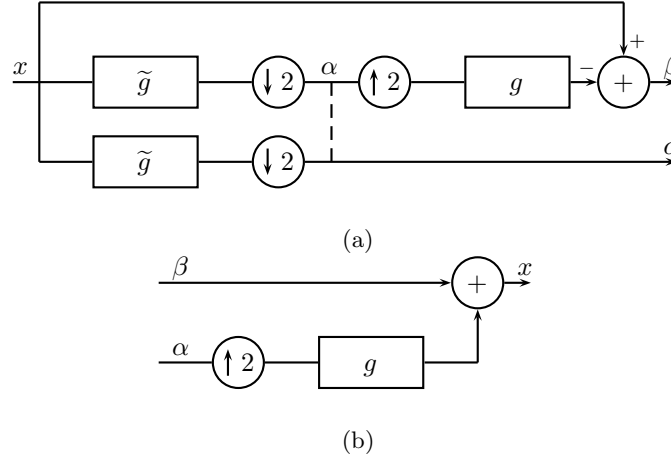
$$\Phi = \{\varphi_{i,n-8k}\}_{k \in \mathbb{Z}, i=0, 1, \dots, 15}. \quad (10.94)$$

Compare this to the DWT example from Section 9.1.

### 10.5.2 Pyramid Frames

Pyramid frames were introduced for coding in 1983 by Burt and Adelson. Although redundant, the pyramid coding scheme was developed for compression of images and was recognized in the late 1980s as one of the precursors of wavelet octave-band decompositions. The scheme works as follows: First, a coarse approximation  $\alpha$  is

<sup>145</sup>Remember that superscript ( $\ell$ ) denotes the level in the tree.



**Figure 10.19:** The (a) analysis and (b) synthesis part of the pyramid filter bank. This scheme implements a frame expansion. The dashed line indicates the actual implementation, as in reality, the lowest branch would not be implemented; it is indicated here for clarity and parallelism with two-channel filter banks.

derived (an example of how this could be done is in Figure 10.19).<sup>146</sup> Then, from this coarse version, the original is predicted (in the figure, this is done by upsampling and filtering) followed by calculating the prediction error  $\beta$ . If the prediction is good (as is the case for most natural images that have a lowpass characteristic), the error will have a small variance and can thus be well compressed. The process can be iterated on the coarse version. The outputs of the analysis filter bank are:

$$\alpha(z) \stackrel{(a)}{=} \frac{1}{2} \left[ \tilde{G}(z^{1/2})X(z^{1/2}) + \tilde{G}(-z^{1/2})X(-z^{1/2}) \right], \quad (10.95a)$$

$$\begin{aligned} \beta(z) &\stackrel{(b)}{=} X(z) - \frac{1}{2}G(z) \left[ \tilde{G}(z)X(z) + \tilde{G}(-z)X(-z) \right] \\ &= X(z) - G(z)\alpha(z^2), \end{aligned} \quad (10.95b)$$

where (a) follows from (2.196a) and (b) from (7.77a). To reconstruct, we simply upsample and interpolate the prediction  $\alpha(z)$  and add it back to the prediction error  $\beta(z)$ :

$$G(z)\alpha(z^2) + \beta(z) = X(z). \quad (10.96)$$

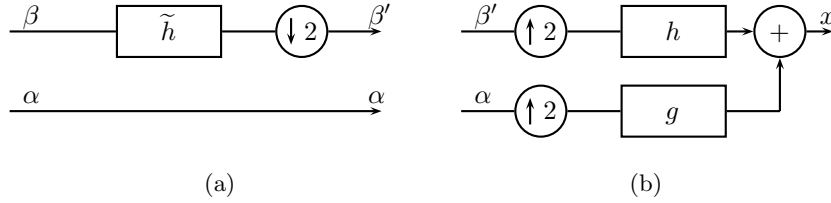
Upsampling and interpolating is, however, only one way to obtain the prediction back at full resolution; any appropriate operator (even a nonlinear one) could have been simply inverted by subtraction. We can also see that in the figure, the redundancy of the system is 50%;  $\alpha$  is at half resolution while  $\beta$  is at full resolution, that is, after analysis, we have 50% more samples than we started with. With the analysis given in Figure 10.19(a), we now have several options:

<sup>146</sup>While in the figure the intensity of the coarse approximation  $\alpha$  is obtained by linear filtering and downsampling, this need not be so; in fact, one of the powerful features of the original scheme is that any operator can be used, not necessarily linear.



## 10.5. Wavelet Frames

761



**Figure 10.20:** The pyramid filter bank implementing a basis expansion. With  $\{g, \tilde{g}, h, \tilde{h}\}$  a biorthogonal set, the scheme implements a biorthogonal basis expansion, while with  $g$  and  $\tilde{g}$  orthogonal, that is,  $\tilde{g}_n = g_{-n}$  and  $g$  satisfies (7.13), the scheme implements an orthonormal basis expansion. The output  $\beta$  from Figure 10.19(a) goes through (a) filtering and downsampling creating a new output  $\beta'$ . (b) Synthesis part.

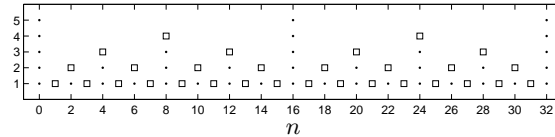
- Synthesis is performed by upsampling and interpolating  $\alpha$  by  $g$  as in Figure 10.19(b). In this case, the resulting scheme is clearly redundant, as we have just discussed, and implements a *frame expansion*, which can be either:
  - (i) *general*, when filters  $g$  and  $\tilde{g}$  are biorthogonal (they satisfy (7.66)), or,
  - (ii) *tight*, when filters  $g$  and  $\tilde{g}$  are orthogonal, that is,  $\tilde{g}_n = g_{-n}$  and  $g$  satisfies (7.13). We illustrate this case in Example 10.9.
- The analysis goes through one more stage, as in Figure 10.20(a), and synthesis is performed as in Figure 10.20(b). In this case, the scheme implements a *basis expansion*, which can be either (both are illustrated in Exercise 10.11):
  - (i) *biorthogonal*, when filters  $g$  and  $\tilde{g}$  are biorthogonal, or,
  - (ii) *orthonormal*, when filters  $g$  and  $\tilde{g}$  are orthogonal.

**EXAMPLE 10.9** We use the pyramid filter bank as in Figure 10.19. Let us assume that  $g$  is the Haar lowpass filter from (7.1a) and that  $\tilde{g}_n = g_{-n}$ . Then we know from Chapter 7, that  $\beta$  is nothing else but the output of the highpass branch, given in (I.9b). For every two input samples, while  $\alpha$  produces one output sample,  $\beta$  produces two output samples; thus, the redundancy. We can write this as:

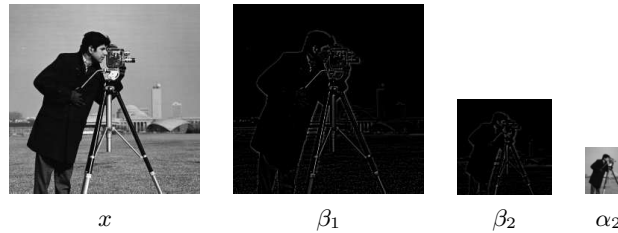
$$\begin{bmatrix} \alpha_n \\ \beta_{2n} \\ \beta_{2n+1} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix}}_{\tilde{\Phi}^T} \begin{bmatrix} x_{2n} \\ x_{2n+1} \end{bmatrix}.$$

We know, however, from our previous discussion that the above matrix is the dual frame matrix  $\tilde{\Phi}^T$ . Finding its canonical dual, we get that  $\Phi = \tilde{\Phi}$ , and thus, this pyramid scheme implements a tight frame expansion.

The redundancy for pyramid frames is  $A_1 = 3/2$  at level 1,  $A_2 = 7/4$  at level 2, leading to  $A_\infty = 2$  (see Figure 10.21), far less than the shift-invariant DWT construction we will see in a moment. Thanks to this constant redundancy,



**Figure 10.21:** Sampling grid corresponding to the time-frequency tiling of the pyramid coding scheme (points—nonredundant, squares—redundant).



**Figure 10.22:** Two-level pyramid decomposition of an image  $x$ . A first-level coarse approximation  $\alpha_1$  is computed. A first-level prediction error  $\beta_1$  is obtained as the difference of  $x$  and the prediction calculated on  $\alpha_1$ . A second-level coarse approximation  $\alpha_2$  is computed. A second-level prediction error  $\beta_2$  is obtained as the difference of  $\alpha_1$  and the prediction calculated on  $\alpha_2$ . The scheme is redundant, as the total number of samples in expansion coefficients  $\beta_1, \beta_2, \alpha_2$  is  $(1 + 1/4 + 1/16)$  times the number original image samples, yielding redundancy of about 31%.

pyramid coding has been used together with directional coding to form the basis for nonseparable multidimensional frames called contourlets (see *Further Reading*). An example of a pyramid decomposition of an image is given in Figure 10.22.

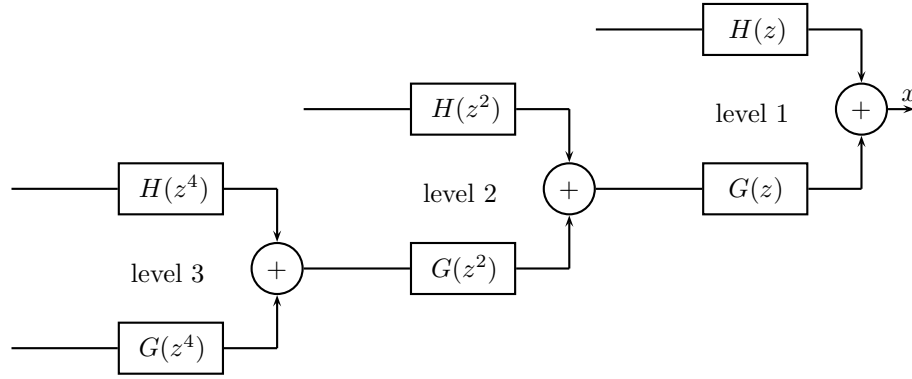
### 10.5.3 Shift-Invariant DWT

The shift-invariant DWT is basically the nondownsampling DWT (an example for  $J = 3$  levels is shown in Figure 10.23). It is sometimes called *stationary wavelet transform*, or, *algorithme à trous*,<sup>147</sup> due to its implementation algorithm by the same name (see Section 10.6.1).

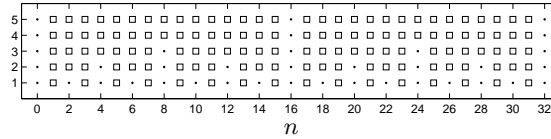
Let  $g$  and  $h$  be the filters used in this filter bank. At level  $\ell$  we will have equivalent upsampling by  $2^\ell$ , which means that the filter moved across the upsampler will be upsampled by  $2^\ell$ , inserting  $(2^\ell - 1)$  zeros between every two samples and thus creating holes (thus *algorithm with holes*).

Figure 10.24 shows the sampling grid for the shift-invariant DWT, from where it is clear that this scheme is completely redundant, as all points are computed. This is in contrast to a completely nonredundant scheme such as the DWT shown in Figure 10.16(a). In fact, while the redundancy per level of this algorithm grows exponentially since  $A_1 = 2, A_2 = 4, \dots, A_J = 2^J, \dots$ , the total redundancy for  $J$  levels is linear, as  $A = A_J 2^{-J} + \sum_{\ell=1}^J A_\ell 2^{-\ell} = (J+1)$ . This growing redundancy is

<sup>147</sup>From French for *algorithm with holes*, coming from the computational method that can take advantage of upsampled filter impulse responses, discussed in Section 10.6.1.



**Figure 10.23:** The synthesis part of the equivalent 3-channel filter bank implementing the shift-invariant DWT with  $J = 3$  levels. The analysis part is analogous and filters are given in (10.92). This is the same scheme as in Figure 10.18 with all the upsamplers removed.



**Figure 10.24:** Sampling grid corresponding to the time-frequency tiling of the shift-invariant DWT (points—nonredundant, squares—redundant).

the price we pay for shift invariance as well as the simplicity of the algorithm. The 2D version of the algorithm is obtained by extending the 1D version in a separable manner, leading to the total redundancy of  $A = A_J 2^{-J} + 3 \sum_{\ell=1}^J A_\ell 2^{-\ell} = (3J+1)$ . Exercise 10.12 illustrates the redundancy of such a frame.

## 10.6 Computational Aspects

### 10.6.1 The Algorithm à Trous

This algorithm was introduced as a fast implementation of the dyadic (continuous) wavelet transform by Holschneider, Kronland-Martinet, Morlet, and Tchamitchian in 1989, and corresponds to the DWT with samplers removed. We introduced it in Section 10.5 as shift-invariant DWT and showed an example for  $J = 3$  in Figure 10.23. The equivalent filters in each branch are computed first, and then, the samplers are removed. Because the equivalent filters are convolutions with upsampled filters, the algorithm can be efficiently computed due to *holes* produced by upsampling.

---

**aTrous**( $\alpha^{(0)}$ )  
**Input:**  $x = \alpha^{(0)}$ , the input signal.  
**Output:**  $\alpha^{(J)}, \beta^{(\ell)}, \ell = 1, 2, \dots, J$ , transform coefficients.

---

```

initialize
for  $\ell = 1$  to  $J$  do
   $\alpha^{(\ell)} = \alpha^{(\ell-1)} * (\uparrow 2^{\ell-1})g$ 
   $\beta^{(\ell)} = \alpha^{(\ell-1)} * (\uparrow 2^{\ell-1})h$ 
end for
return  $\alpha^{(J)}, \beta^{(\ell)}, \ell = 1, 2, \dots, J$ 

```

**Table 10.5:** Algorithm *à trous* implementing the shift-invariant DWT. Upsampling an impulse response  $g$  by a factor of  $n$  is denoted by  $(\uparrow n)g$ .

## 10.6.2 Efficient Gabor and Spectrum Computation

## 10.6.3 Efficient Sparse Frame Expansions

### Matching Pursuit

### Orthonormal Matching Pursuit

### Linear Programming

## Chapter at a Glance

This chapter relaxed the constraint of nonredundancy bases carry, using frames to achieve robustness and freedom in choosing not only the *best expansion*, but also, given a fixed expansion, the *best expansion coefficients* under desired constraints. We introduced these mostly on finite-dimensional frames, as they can be easily visualized via rectangular matrices. The infinite-dimensional frames we discussed were only those implementable by oversampled filter banks, summarized in Table 10.6.

Oversampled Filter Bank			
<b>Block diagram</b>			
<b>Basic characteristics</b>			
number of channels	$M > N$		
sampling factor	$N$		
channel sequences	$\alpha_{i,n}$	$i = 0, 1, \dots, M - 1$	
<b>Filters</b>		<b>Synthesis Analysis</b>	
filter $i$	$g_{i,n}$	$\tilde{g}_{i,n}$	$i = 0, 1, \dots, M - 1$
polyphase component $j$	$g_{i,j,n}$	$\tilde{g}_{i,j,n}$	$j = 0, 1, \dots, N - 1$

**Table 10.6:** Oversampled filter bank.

	Block transforms (Fourier-like)	Overlapped transforms	Time-frequency constraints (wavelet-like)
Bases	DFT	LOT	DWT
Frames	HTF	Local Fourier	Oversampled DWT

**Table 10.7:** Bases versus frames.

We discussed two big classes of frames following their counterparts in bases: local Fourier frames and wavelet frames. Table 10.7 depicts relationships existing between various classes of bases and frames. For example, the block-transform counterpart of the DFT are the harmonic tight frames, while the same for the LOT will be local Fourier frames, obtained by both complex-exponential modulation as well as cosine modulation. By increasing the support of basis functions we can go from the DFT to the LOT, and similarly,

from harmonic tight frames to local Fourier frames. Imposing time-frequency constraints leads to new classes of representations, such as the DWT, whose frame counterpart is the oversampled DWT.

## Historical Remarks

In the signal processing and harmonic analysis communities, frames are generally considered to have been born in 1952 in the paper by Duffin and Schaeffer [50]. Despite being over half a century old, frames gained popularity only in the 1990s, due mostly to the work of three wavelet pioneers—Daubechies, Grossman and Meyer [42]. An important piece to understanding frames came with Naimark's theorem, known for a long time in operator algebra and used in quantum information theory, and rediscovered by several people in the 1990s, among others, Han and Larson [67]; they came up with the idea that a frame could be obtained by compressing a basis in a larger space.

The idea behind the class of complex exponential-modulated frames, consisting of many families, dates back to Gabor [55] with insight of constructing bases by modulation of a single prototype function. Gabor originally used complex-exponential modulation, and thus, all those families with the same type of modulation are termed *complex exponential-modulated frames*, or sometimes, Gabor frames. Other types of modulation are possible, such as cosine modulation, and again, all those families with cosine modulation are termed *cosine-modulated frames*.

Frame-like ideas, that is, building redundancy into a signal expansion, can be found in numerous fields, from source and channel coding, to communications, classification, operator and quantum theory.

## Further Reading

**Books and Textbooks** The sources on frames are the book by Daubechies [41], a text by Christensen [29] a number of classic papers [25, 69, 40, 67] as well as an introductory tutorial on frames by Kovačević and Chebira [90].

**Results on Frames** A thorough analysis of oversampled filter banks implementing frame expansions is given in [37, 16, 38]. Following up on the result of Benedetto and Fickus [9] on minimizing frame potential from Section 10.2.1, Cassaza, Fickus, Kovačević, Leon and Tremain extended the result to nonequal-norm tight frames, giving rise to the *fundamental inequality*, which has ties to the capacity region in synchronous CDMA systems [170]. Casazza and Kutyniok in [27] investigated Gram-Schmidt-like procedure for producing tight frames. In [13], the authors introduce a quantitative notion of redundancy through local redundancy and a redundancy function, applicable to all finite-dimensional frames.

**Local Fourier Frames** For finite-dimensional frames, similar ideas to those of harmonic tight frames have appeared in the work by Eldar and Bölcskei [51] under the name *geometrically uniform frames*, frames defined over a finite Abelian group of unitary matrices both with a single as well as multiple generators. Harmonic tight frames have been generalized in the works by Vale and Waldron [163], as well as Casazza and Kovačević [26].

Harmonic tight frames, as well as equiangular frames (where  $|\langle \varphi_i, \varphi_j \rangle|$  is a constant) [119], have strong connections to *Grassmannian frames*. In a comprehensive paper [146], Strohmer and Heath discuss those frames and their connection to Grassmannian packings, spherical codes, graph theory and Welch Bound sequences (see also [75]).

The lack of infinite-dimensional bases with good time and frequency localization, the result of the discrete Balian-Low theorem, prompted the development of oversampled filter banks that use complex-exponential modulation. They are known under various names: *oversampled DFT filter banks*, *complex exponential-modulated filter banks*, *short-time Fourier filter banks* and *Gabor filter banks* and have been studied in [36, 16, 15, 14, 53, 145]. Bölcskei and Hlawatsch in [15] have studied the other type of modulation, cosine. The connection between these two classes is deep as there exists a general decomposition of the frame operator corresponding to a cosine-modulated filter bank as the sum of the frame operator of the underlying complex exponential-modulated frame and an additional operator, which vanishes under certain conditions [7]. The *lapped tight frame transforms* were proposed as a way to obtain a large number of frames by seeding from LOTs [28, 125].

**Wavelet Frames** Apart from those already discussed, like pyramid frames [21], many other wavelet-like frame families have been proposed, among them, the *dual-tree complex wavelet transform*, a nearly shift-invariant transform with redundancy of only 2, introduced by Kingsbury [85, 86, 87]. Selesnick in [127, 128] followed with the *double-density DWT* and variations, which can approximately be implemented using a 3-channel filter bank with sampling by 2, again nearly shift invariant with redundancy that tends towards 2 when iterated. Some other variations include *power-shiftable DWT* [132] or *partial DWT* [137], which removes samplers at the first level but leaves them at all other levels, with redundancy  $A_j = 2$  at each level and again near shift invariance. Bradley in [18] introduces the *overcomplete DWT*, the DWT with critical sampling for the first  $k$  levels followed by the shift-invariant DWT for the last  $j - k$  levels.

**Multidimensional Frames** Apart from obvious, tensor-like, constructions of multidimensional frames, true multidimensional solutions exist. The oldest multidimensional frame seems to be the *steerable pyramid* introduced by Simoncelli, Freeman, Adelson and Heeger in 1992 [132], following on the previous work by Burt and Adelson on pyramid coding [21]. The steerable pyramid possesses many nice properties, such as joint space-frequency localization, approximate shift invariance, approximate tightness and approximate rotation invariance. An excellent overview of the steerable pyramid and its applications is given on Simoncelli's web page [131].

Another multidimensional example is the work of Do and Vetterli on *contourlets* [44, 35], motivated by the need to construct efficient and sparse representations of intrinsic geometric structure of information within an image. The authors combine the ideas of pyramid filter banks [43] with directional processing, to obtain contourlets, expansions capturing contour segments. These are almost critically sampled, with redundancy of 1.33.

Some other examples include [97] where the authors build both critically-sampled and shift-invariant 2D DWT. Many "-lets" are also multidimensional frames, such as *curvelets* [24, 23] and *shearlets* [94]. As the name implies, curvelets are used to approximate curved singularities in an efficient manner [24, 23]. As opposed to wavelets, which use dilation and translation, shearlets use dilation, shear transformation and translation, and possess useful properties such as directionality, elongated shapes and many others [94].

**Applications of Frames** Frames have become extremely popular and have been used in many application fields. The text by Kovačević and Chebira [90] contains an overview of many of these and a number of relevant references. In some fields, frames have been used for years, for example in CDMA systems, in the work of Massey and Mittelholzer [104] on

Welch bound and sequence sets for CDMA systems. It turns out that the Welch bound is equivalent to the frame potential minimization inequality. The equivalence between unit-norm tight frames and Welch bound sequences was shown in [146]. Waldron formalized that equivalence for general tight frames in [171], and consequently, tight frames are referred in some works as Welch bound sequences [150].

## Exercises with Solutions

### 10.1. Expansion of a Complex Exponential and a Kronecker Delta

Given is a sequence  $x$  consisting of a single complex sinusoidal sequence of unknown frequency  $(2\pi/N)\ell$  and a Kronecker delta sequence of unknown location  $k$ :

$$x_n = \beta_1 e^{j(2\pi/N)\ell n} + \beta_2 \delta_{n-k}$$

for  $n = 0, 1, \dots, N-1$ , and  $\ell, k \in [0, 1, \dots, N-1]$ . Given a frame consisting of a DFT basis and an identity basis as in (10.28), compute the expansion coefficients  $\alpha = \Phi^* x$ , which, as per (10.29), lead to perfect reconstruction  $x = (1/2)\Phi\alpha$ .

*Solution:* Consider expansion coefficients  $\alpha_i = \langle x, \varphi_i \rangle$ .

If  $\varphi_i$  is one of the DFT basis vectors (2.160), that is,  $i = 0, 1, \dots, N-1$ , then

$$\begin{aligned} \alpha_i &= \sum_{n=0}^{N-1} x_n \varphi_{i,n}^* \\ &= \sum_{n=0}^{N-1} (\beta_1 e^{j(2\pi/N)\ell n} + \beta_2 \delta_{n-k}) \left( \frac{1}{\sqrt{N}} e^{-j(2\pi/N)in} \right) \\ &= \frac{\beta_1}{\sqrt{N}} \sum_{n=0}^{N-1} e^{j(2\pi/N)(\ell-i)n} + \frac{\beta_2}{\sqrt{N}} e^{-j(2\pi/N)ik} \\ &\stackrel{(a)}{=} \begin{cases} \beta_1 \sqrt{N} + \frac{\beta_2}{\sqrt{N}} e^{-j(2\pi/N)\ell k}, & i = \ell; \\ \frac{\beta_2}{\sqrt{N}} e^{-j(2\pi/N)ik}, & i \neq \ell; \end{cases} \end{aligned}$$

where (a) follows from the orthogonality of the roots of unity (2.277c). Therefore, among the first  $N$  expansion coefficients, if  $N\beta_1 \gg \beta_2$ , only one will stand out, namely, that matching the frequency  $(2\pi/N)\ell$ .

If, on the other hand,  $\varphi_i$  is one of the standard basis vectors, that is,  $i = N, N+1, \dots, 2N-1$ , then

$$\begin{aligned} \alpha_i &= \sum_{n=0}^{N-1} x_n \varphi_{i,n}^* \\ &= \sum_{n=0}^{N-1} (\beta_1 e^{j(2\pi/N)\ell n} + \beta_2 \delta_{n-k}) \delta_{n+N-i} \\ &= \beta_1 \sum_{n=0}^{N-1} e^{j(2\pi/N)\ell n} \delta_{n+N-i} + \beta_2 \sum_{n=0}^{N-1} \delta_{n-k} \delta_{n+N-i} \\ &\stackrel{(a)}{=} \begin{cases} \beta_1 e^{j(2\pi/N)\ell k} + \beta_2, & i = N+k; \\ \beta_1 e^{j(2\pi/N)\ell i}, & i \neq N+k. \end{cases} \end{aligned}$$

Thus, among the last  $N$  expansion coefficients, if  $\beta_2 \gg \beta_1$ , only one will stand out, the coefficient matching the location of the Kronecker delta impulse,  $k$ . In other words, if we



choose  $\beta_1 \sim O(1)$  and  $\beta_2 \sim O(N)$ , the two coefficients localizing the complex sinusoid and the Kronecker delta impulse will stand out. Note, however, that while this is a minimum  $\ell^2$ -norm solution, it is not sparse. It is possible to find an expansion coefficient vector  $\alpha'$  with exactly 2 nonzero coefficients; it will be sparse, with a larger  $\ell^2$ -norm.

10.2. *Canonical Dual Frame Minimizes  $\ell^2$  Norm of an Expansion*

Given a frame  $\Phi$  and its canonical dual frame (10.65a), show that among all possible dual frames, the canonical dual frame minimizes  $\|\alpha\|^2$ .

*Solution:* The solution uses the geometry of the expansion as we have seen in Section 10.2. The frame  $\Phi$  is of size  $N \times M$  and  $\text{rank}(\Phi) = N$ . Moreover, its null space is of size  $(M - N)$  according to Table 1.2. Call  $\tilde{\varphi}_i^\perp$ ,  $i = 0, 1, \dots, M - N - 1$ , the vectors spanning that null space, that is,  $\Phi \tilde{\varphi}_i^\perp = 0$ , for  $i = 0, 1, \dots, M - N - 1$ . Call  $\tilde{\Phi}^\perp$  the following matrix of size  $N \times M$ :

$$\tilde{\Phi}^\perp = [\tilde{\varphi}_0^\perp \quad \tilde{\varphi}_1^\perp \quad \dots \quad \tilde{\varphi}_{M-N-1}^\perp] \Gamma,$$

where  $\Gamma$  is an arbitrary, rank- $(M - N)$ ,  $(M - N) \times M$  matrix of scalars. In other words,  $\tilde{\Phi}^\perp$  is just a matrix of  $M$  vectors spanning the null space of  $\Phi$  (linear combinations of  $\tilde{\varphi}_i^\perp$ 's). We thus know that

$$\Phi(\tilde{\Phi}^\perp)^* = 0,$$

for any  $\Gamma$ . This also means that any dual of  $\Phi$  can be written as

$$\tilde{\Phi} + \tilde{\Phi}^\perp,$$

since

$$\Phi(\tilde{\Phi} + \tilde{\Phi}^\perp)^* = \underbrace{\Phi\tilde{\Phi}^*}_I + \underbrace{\Phi(\tilde{\Phi}^\perp)^*}_0 = I.$$

We can now express the expansion coefficients as a function of  $\tilde{\Phi}^\perp$  as

$$\alpha(\tilde{\Phi}^\perp) = (\tilde{\Phi}^* + (\tilde{\Phi}^\perp)^*)x.$$

To minimize its energy, we have to minimize

$$\begin{aligned} \|\alpha(\tilde{\Phi}^\perp)\|^2 &= \alpha(\tilde{\Phi}^\perp)^* \alpha(\tilde{\Phi}^\perp) \\ &= x^* (\tilde{\Phi} + \tilde{\Phi}^\perp) (\tilde{\Phi}^* + (\tilde{\Phi}^\perp)^*) x \\ &= x^* (\tilde{\Phi}\tilde{\Phi}^* + \tilde{\Phi}(\tilde{\Phi}^\perp)^* + \tilde{\Phi}^\perp\tilde{\Phi}^* + \tilde{\Phi}^\perp(\tilde{\Phi}^\perp)^*) x, \end{aligned}$$

a quadratic form minimized for  $\tilde{\Phi}^\perp = 0$  (actually  $\Gamma = 0$ ).

10.3. *Computation of the Canonical Dual Frame*

Given a frame  $\Phi$  and its canonical dual frame (10.65a), show that it can be computed as:

$$\tilde{\Phi} = \frac{2}{\lambda_{\min} + \lambda_{\max}} \sum_{k=0}^{\infty} \left( I - \frac{2}{\lambda_{\min} + \lambda_{\max}} T \right)^k \Phi.$$

*Solution:* We just sketch a proof here; see [41] for a rigorous development.

If frame bounds  $\lambda_{\min}$  and  $\lambda_{\max}$  are close, that is, if

$$\nabla = \kappa(T) - 1 \ll 1,$$

then (10.62) implies that  $T$  is close to  $((\lambda_{\min} + \lambda_{\max})/2)I$ , or  $T^{-1}$  is close to  $(2/(\lambda_{\min} + \lambda_{\max}))I$ . This further means that  $x$  can be written as follows:

$$x = \frac{2}{\lambda_{\min} + \lambda_{\max}} \sum_{i=0}^{M-1} \langle x, \varphi_i \rangle \varphi_i + Rx,$$

where  $R$  is given by (use (10.64b))

$$R = I - \frac{2}{\lambda_{\min} + \lambda_{\max}} T. \quad (\text{E10.3-1})$$

Using (10.62) we obtain

$$-\frac{\kappa(T)-1}{\kappa(T)+1}I \leq R \leq \frac{\kappa(T)-1}{\kappa(T)+1}I,$$

and, as a result,

$$\|R\| \leq \frac{\kappa(T)-1}{\kappa(T)+1} = \frac{\nabla}{2+\nabla} \leq 1. \quad (\text{E10.3-2})$$

From (E10.3-1) and (E10.3-2),  $T^{-1}$  can be written as

$$T^{-1} = \frac{2}{\lambda_{\min} + \lambda_{\max}}(I - R)^{-1} = \frac{2}{\lambda_{\min} + \lambda_{\max}} \sum_{i=0}^{\infty} R^i,$$

implying that

$$\begin{aligned} \tilde{\varphi}_i &= T^{-1}\varphi_i = \frac{2}{\lambda_{\min} + \lambda_{\max}} \sum_{k=0}^{\infty} R^k \varphi_i \\ &= \frac{2}{\lambda_{\min} + \lambda_{\max}} \sum_{k=0}^{\infty} \left( I - \frac{2}{\lambda_{\min} + \lambda_{\max}} T \right)^k \varphi_i. \end{aligned}$$

Like for biorthogonal bases in Section 1.5.3, we can characterize the convergence using the condition number of the Hermitian matrix  $T$ ,  $\kappa(T) = \lambda_{\max}/\lambda_{\min}$ . Thus, when  $\kappa(T)$  is large, convergence is slow, while as  $\kappa(T)$  tends to 1, convergence is faster, and the dual frame is close to  $\Phi$ . Specifically, when all the vectors are of unit norm and  $\lambda_{\min} = \lambda_{\max} = 1$ , we have an orthonormal basis and  $T^{-1} = I$ .

#### 10.4. Condition Numbers and Convergence

Compute the condition number and comment on the convergence of  $\tilde{\Phi}$  for the following three frames:

- (i) Frame given in (10.19).
- (ii) Frame given in (10.13), where the first vector is perturbed by  $\theta$ , so that the resulting frame is now

$$\Phi = \begin{bmatrix} \cos \theta & -\frac{1}{2} & -\frac{1}{2} \\ \sin \theta & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}. \quad (\text{E10.4-1})$$

- (iii) Frame given by

$$\Phi = \begin{bmatrix} 1 & \cos \theta & \cos 2\theta \\ 0 & \sin \theta & \sin 2\theta \end{bmatrix}. \quad (\text{E10.4-2})$$

*Solution:*

- (i) We start with (10.19). We have already computed the eigenvalues of  $\tilde{\Phi}\tilde{\Phi}^*$ . It turns out that  $T = \Phi\Phi^* = \tilde{\Phi}\tilde{\Phi}^*$ , and thus, those eigenvalues are the same,  $\lambda_{\min} = 1$ ,  $\lambda_{\max} = 3$ . The convergence depends on  $\kappa(T) = 3$ , meaning that it is reasonable but not very close to one.

Turning this frame into a unit-norm one (see Exercise 10.6), we get a new frame

$$\Phi' = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\frac{1}{\sqrt{2}} \\ 0 & 1 & -\frac{1}{\sqrt{2}} \end{bmatrix},$$

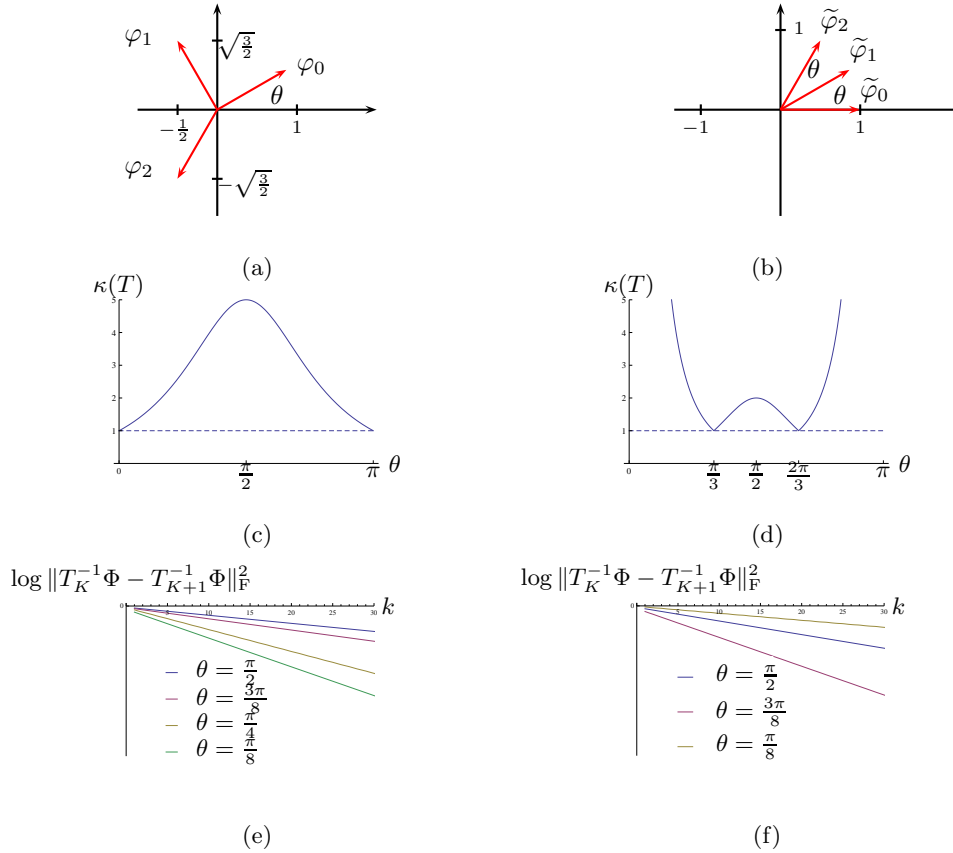
with new frame bounds

$$\lambda'_{\min} = \frac{1}{2}(3 - \sqrt{5}) \quad \lambda'_{\max} = \frac{1}{2}(3 + \sqrt{5}),$$

and new condition number of

$$\kappa(T') = \frac{3 + \sqrt{5}}{3 - \sqrt{5}} \sim 6.85,$$

more than twice as large as for the unnormalized version where  $\kappa(T) = 3$ .



**Figure E10.4-1:** Two example frames, (a) from (E10.4-1) and (b) from (E10.4-2), and the behavior of their respective  $\kappa(T)$  in (c) and (d). In (e) and (f) we show the convergence behavior of the Frobenius norm of the difference between the true dual frame and that computed using (10.67) for (c) and (d), respectively.

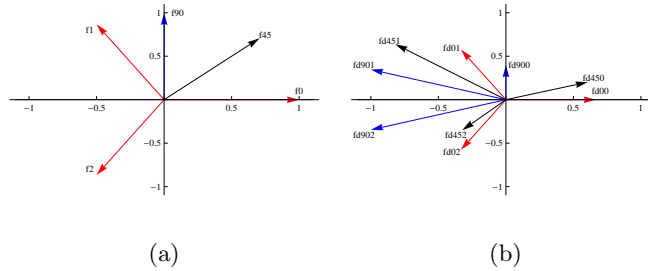
- (ii) This frame is given in Figure E10.4-1(a). The eigenvalues of the corresponding  $T$  are

$$\lambda_{\min} = \frac{3}{2} - |\sin \theta| \quad \lambda_{\max} = \frac{3}{2} + |\sin \theta|,$$

$$\kappa(T) = \frac{3 + 2|\sin \theta|}{3 - 2|\sin \theta|}.$$

Figure E10.4-1(c) illustrates the behavior of the ratio  $\kappa(T)$ . For  $\theta = 0$ , we get our original, unperturbed, frame back, and thus,  $\kappa(T) = 1$ , meaning that  $T = (3/2)I$  and no inversion is necessary to compute the dual frame. When  $\theta = \pi/2$ ,  $\kappa(T)$  reaches its maximum of 5, when the perturbed  $\varphi_0$  finds itself between  $\varphi_1$  and  $-\varphi_2$ . The frame is not well behaved and (10.67) will take a long time to converge. The ratio  $\kappa(T)$  then falls back to 1 for  $\theta = \pi$ , as then the perturbed  $\varphi_0$  is just  $-\varphi_0$ . Figure E10.4-1(e) illustrates convergence for a few angles  $\theta$ . In the figure,  $T_K^{-1}$  denotes  $T^{-1}$  computed via (10.67) using  $(K+1)$  terms of the summation. Convergence is measured using the logarithm of the difference, in the squared Frobenius norm (see (1.217)), of the

dual frame operator between two consecutive iterations. We see that, as expected, convergence is the fastest for  $\theta = \pi/8$  (smallest  $\kappa(T)$ ) and decreases as  $\theta$  increases to  $\pi/2$ . Figure E10.4-2 illustrates the behavior of the dual frame graphically for three values of  $\theta = 0, \pi/4, \pi/2$ ; we can see how the dual frame degenerates from the dual frame for  $\theta = 0$  (basically the frame itself, just scaled), through  $\theta = \pi/4$ , and finally to  $\theta = \pi/2$ , where the first vector is very small and the other two are very close to being colinear.



**Figure E10.4-2:** Three different frames from Figure E10.4-1(a) for  $\theta = 0, \pi/4, \pi/2$ . (a) Frame vector  $\varphi_0$  for  $\theta = 0$  (red),  $\theta = \pi/4$  (black) and  $\theta = \pi/2$  (blue). Frame vectors  $\varphi_1, 2$  are the same for all three frames. (b) Corresponding dual frames for  $\theta = 0$  (red),  $\theta = \pi/4$  (black) and  $\theta = \pi/2$  (blue).

- (iii) This frame is given in Figure E10.4-1(b). The eigenvalues of the corresponding  $T$  are

$$\lambda_{\min} = \frac{1}{2} (3 - |2 \cos 2\theta + 1|) \quad \lambda_{\max} = \frac{1}{2} (3 + |2 \cos 2\theta + 1|)$$

$$\kappa(T) = \frac{3 + |2 \cos 2\theta + 1|}{3 - |2 \cos 2\theta + 1|}.$$

Figure E10.4-1(d) illustrates the behavior of the ratio  $\kappa(T)$ . For  $\theta = \pi/3$  and  $\theta = 2\pi/3$ , the ratio is 1 and no inversion is necessary. Note that for  $\theta = 2\pi/3$  we have exactly the same frame we just saw (unperturbed version), the same one from (10.13). For  $\theta \in [\pi/3, 2\pi/3]$ , the ratio is between 1 and 2, with  $\kappa(T) = 2$  for  $\theta = \pi/2$ . However, for  $\theta < \pi/3$  and  $\theta > 2\pi/3$ , the ratio becomes unbounded. This is easily understood as we think what would happen around  $\theta = 0$ . All three frame vectors are identical (or very close to each other), and thus, the frame would disintegrate to a single vector. Figure E10.4-1(f) illustrates convergence for a few angles  $\theta$ . We see that, as expected, convergence is slower for  $\theta = \pi/2$  than for  $\theta = 3\pi/8$  (larger  $\kappa(T)$ ) and the slowest for  $\theta = \pi/8$ .

These simple examples illustrated the meaning behind the frame bounds  $\lambda_{\min}$ ,  $\lambda_{\max}$  and the behavior of their ratio  $\kappa(T)$ .

## Exercises

### 10.1. Unit-Norm Tight Frames

Prove that the following two statements are equivalent:

- $\{\varphi_i = [\cos \theta_i \quad \sin \theta_i]^T\}_{i=0}^{M-1}$ , is a unit-norm tight frame.
- $\sum_{i=0}^{M-1} z_i = 0$  where  $z_i = e^{j2\theta_i}$  for  $i = 0, 1, \dots, M-1$ .

### 10.2. Parametrization of Finite-Dimensional Frames

Given is a unit-norm tight frame with 4 vectors in  $\mathbb{R}^2$ ,  $\{\varphi_i = [\cos \theta_i \quad \sin \theta_i]^T\}_{i=0}^3$ . Prove

that all unit-norm tight frames with  $N = 2$ ,  $M = 4$ , are unions of two orthonormal bases, parameterized by the angle between them (within the equivalence class of all frames obtained from the original frame by rigid rotations, reflections around an axis and negation of individual vectors).

(Hint: Use the result of Exercise 10.1.)

### 10.3. Harmonic Tight Frames

Harmonic tight frames are obtained from the DFT by seeding, that is, by deleting the last  $(M - N)$  columns of the DFT matrix  $\Psi = \text{DFT}_M$  from (2.161a) yielding the harmonic tight frame matrix  $\Phi$  as in (10.47a).

- (i) Show that a harmonic tight frame is indeed tight and find its frame bound. Find its dual frame  $\tilde{\Phi}$ .
- (ii) Consider now applying a *running* transform, that is, applying  $\Psi^*$  on an input of length  $N$  and advancing it by  $N$  samples at a time. Choose  $M = 4$ ,  $N = 2$ . Find  $\Phi$ ,  $\Phi^*$ , and  $\tilde{\Phi}$  and comment.

### 10.4. Vandermonde Frames

Given is a frame with  $M$  vectors in an  $N$ -dimensional space generated by  $[1 \quad t_1 \quad t_2 \quad \dots \quad t_{M-1}]$ ,  $t_i \neq t_j$ , as follows:

$$\Phi = [\varphi_0 \quad \varphi_1 \quad \dots \quad \varphi_{M-1}] = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & t_1 & t_2 & \dots & t_{M-1} \\ 1 & t_1^2 & t_2^2 & \dots & t_{M-1}^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & t_1^{N-1} & t_2^{N-1} & \dots & t_{M-1}^{N-1} \end{bmatrix}.$$

- (i) Prove that any subset of  $N$  frame vectors is linearly independent, and therefore, that  $\Phi$  contains  $\binom{M}{N}$  different biorthogonal bases. Frames with these property are said to be *maximally robust to erasures*, as in Definition 10.4.
- (ii) Choose  $t_i = W_M^i = e^{-j2\pi i/M}$ , or, a harmonic tight frame as in (10.47b). Show that it satisfies (i), that is, it is maximally robust.

### 10.5. Useful Frame Facts

Prove the following frame facts (10.64):

- (i) (10.64a):  $\sum_{j=0}^{N-1} \lambda_j = \sum_{i=0}^{M-1} \|\varphi_i\|^2$ ;
- (ii) (10.64b):  $Tx = \sum_{i=0}^{M-1} \langle x, \varphi_i \rangle \varphi_i$ ;
- (iii) (10.64c):  $\langle x, Tx \rangle = \sum_{i=0}^{M-1} |\langle x, \varphi_i \rangle|^2$ ;
- (iv) (10.64d)  $\sum_{i=0}^{M-1} \langle \varphi_i, T\varphi_i \rangle = \sum_{i,j=0}^{M-1} |\langle \varphi_i, \varphi_j \rangle|^2$ .

### 10.6. Frame Normalization

Show that we can normalize any frame  $\Phi$  to have all the vectors of unit norm. Find whether or not rank, eigenvalues of  $T$  and frame bounds change due to the normalization.

### 10.7. Frame Expansions with Positive Expansion Coefficients

Prove that to obtain a frame expansion with all positive expansion coefficients, one needs at least  $M = N + 1$  frame vectors in an  $N$ -dimensional space. Find one such example.

(Hint: Consider the tight frame for  $\mathbb{R}^2$  in (10.13) first, and then generalize.)

### 10.8. Oversampled Filter Bank

Given is a 3-channel analysis/synthesis filter bank with sampling by 2 as in Figure 10.4. The channel signals  $\alpha_i$  are filtered by the channel filters  $C_i$ ,  $i = 0, 1, 2$  before going through the upsamplers. The analysis, synthesis and channel filters are given by

$$\begin{aligned} \tilde{G}_0(z) &= z^{-1}, & \tilde{G}_1(z) &= 1 + z^{-1}, & \tilde{G}_2(z) &= 1 \\ G_0(z) &= 1 - z^{-1}, & G_1(z) &= z^{-1}, & G_2(z) &= z^{-2} - z^{-1}, \\ C_0(z) &= F_0(z), & C_1(z) &= F_0(z) + F_1(z), & C_2(z) &= F_1(z). \end{aligned}$$

Verify that the overall system is shift invariant and performs a convolution with the filter  $F(z) = z^{-1}(F_0(z^2) + z^{-1}F_1(z^2))$ .

10.9. *Overcoming Limitations of Balian-Low Theorem*

Mimic the analysis in Example 10.7 and find the conditions for a 4-channel complex exponential-modulated filter bank with sampling by 2 to implement a tight frame expansion.

10.10. *Oversampled Complex Exponential-Modulated Transmultiplexer*

Consider a complex exponential-modulated oversampled filter bank implementing a transmultiplexer: Start with a two-channel synthesis filter bank with upsampling by 4, followed by a two-channel analysis filter bank with downsampling by 4. This is a redundant scheme since only 2 signals are multiplexed onto a channel that has 4 times higher rate.

- (i) Express this system in polyphase domain.
- (ii) Use the fact that the filters are modulated to express the polyphase matrix as the product of a diagonal matrix and  $2 \times 2$  Fourier matrices.
- (iii) Express the input-output relationship, and indicate conditions on  $G(z)$  and  $\tilde{G}(z)$  to obtain perfect reconstruction.
- (iv) Verify that  $G(z) = \tilde{G}(z) = (1/2)(1 + z^{-1} + z^{-2} + z^{-3})$  leads to perfect reconstruction in this redundant case (two-channel synthesis/analysis filter bank with sampling by 4) while it does not for critical sampling (two-channel synthesis/analysis filter bank with sampling by 2).

10.11. *Pyramid Filter Banks Implementing Basis Expansions*

Consider a pyramid decomposition as discussed in Section 10.5.2. The analysis filter bank is shown in Figure 10.19 and Figure 10.20(a), and the synthesis in Figure 10.20(b).

- (i) Assume that filters  $g$  and  $\tilde{g}$  are biorthogonal, that is, they satisfy (7.66), and show perfect reconstruction.
- (ii) Assume that filters  $g$  and  $\tilde{g}$  are orthogonal, that is,  $\tilde{g}_n = g_{-n}$ , and  $g$  satisfies (7.13) and show perfect reconstruction. Verify the analysis by showing outputs at different points in the system with Haar filters from Table 7.8.

10.12. *Shift-Invariant DWT*

Given is a two-channel orthogonal filter bank as in Theorem 7.1.

- (i) Remove samplers from such a filter bank and prove the following energy-conservation equality:

$$\|x\|^2 = 2(\|\alpha\|^2 + \|\beta\|^2),$$

where  $x$  is the input signal and  $\alpha, \beta$  are the sequences of transform coefficients at the outputs of the lowpass/highpass channels, respectively.

- (ii) Show how to construct a shift-invariant DWT from such a filter bank.

10.13. *Cost of Computing with Harmonic Tight Frames*

Given is a harmonic tight frame as in (10.47a) and Exercise 10.3.

- (i) With  $N = M/2$  and the cost of computing the DFT as in (2.261), what is the cost of computing the frame transform of a vector? What about the dual frame?  
(*Hint:* It is enough to consider the straightforward algorithm; you do not have to try to take advantage of overlaps in your computations.)
- (ii) Consider now applying a *running* transform, that is, applying a length- $M = 2^m$  DFT  $\Psi^*$  on an input of length  $N = 2^{m-k}$ ,  $k = 1, 2, \dots, N$ , and advancing it by  $N$  samples at a time. What is the cost per input sample of computing this running transform as a function of  $N$  and  $M$ ?  
(*Hint:* As above, it is enough to consider the straightforward algorithm; you do not have to try to take advantage of overlaps in your computations.)

## Chapter 11

# Local Fourier Transforms, Frames and Bases on Functions

*Dear Reader,*

*This chapter needs to be finished. The only existing section, Section 11.2 has been proofread and integrated with the previous text. The rest of the sections are yet to be written.*

*Please read on.*

— MV, JK, and VKG

The aim of this chapter follows that of Chapter 8, but for functions. We look for ways to localize the analysis Fourier transform provides by windowing the complex exponentials. As before, this will improve the time localization of the corresponding transform at the expense of the frequency localization. The original idea dates back to Gabor, and thus *Gabor transform* is frequently used; *windowed Fourier transform* and *short-time Fourier transform* are as well. We choose the intuitive *local Fourier transform*, as a counterpart to *local Fourier bases* from Chapter 8 and *local Fourier frames* from Chapter 10.

We start with the most redundant one, local Fourier transform, and then sample to obtain local Fourier frames. With critical sampling we then try for local Fourier bases, where, not surprisingly after what we have seen in Chapter 8, bases with simultaneously good time and frequency localization do not exist, the result known as Balian-Low theorem. Again as in Chapter 8, cosine local Fourier bases do exist, as do wavelet ones we discuss in the next chapter.

## 11.1 Introduction

### Fourier Series Basis Expansion

### Localization Properties of the Fourier Series

### Chapter Outline

We start with the most redundant one version of the local Fourier transform, the local Fourier transform in Section 11.2, and then sample to obtain local Fourier

frames in Section 11.3. With critical sampling we then try for local Fourier bases in Section 11.4, where, not surprisingly after what we have seen in Chapter 8, complex exponential-modulated local Fourier bases with simultaneously good time and frequency localization do not exist, the result known as Balian-Low theorem. Again as in Chapter 8, cosine-modulated local Fourier bases do exist, as do wavelet ones we discuss in the next chapter.

*Notation used in this chapter:* The prototype window function in this chapter is named  $p(t)$ ; this is for consistency with the prototype window sequences used in Chapters 8 and 10;  $g(t)$  is more commonly seen in the literature.  $\square$

## 11.2 Local Fourier Transform

Given a function  $x(t)$ , we start with its Fourier transform  $X(\omega)$  as in Definition 3.10. We analyze  $x(t)$  locally by using a prototype window function  $p(t)$ . We will assume  $p(t)$  is symmetric,  $p(t) = p(-t)$ , and real. The prototype function should be smooth as well, in particular, it should be smoother than the function to be analyzed.<sup>148</sup>

### 11.2.1 Definition of the Local Fourier Transform

We can look at our windowing with the prototype function  $p(t)$  in two ways:

- (i) We window the function  $x(t)$  as

$$x_\tau(t) = p(t - \tau)x(t), \quad (11.1)$$

and then take its Fourier transform (3.48a),

$$X_\tau(\omega) = \langle x_\tau, v_\omega \rangle = \int_{t \in \mathbb{R}} x_\tau(t) e^{-j\omega t} dt = \int_{t \in \mathbb{R}} x(t) p(t - \tau) e^{-j\omega t} dt, \quad (11.2)$$

for  $\omega \in \mathbb{R}$ .

- (ii) We window the complex exponentials  $v_\omega(t) = e^{j\omega t}$  yielding

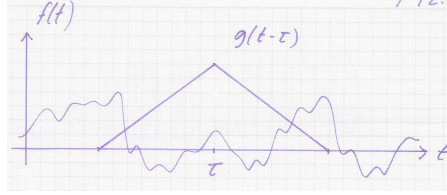
$$\begin{aligned} g_{\Omega, \tau}(t) &= p(t - \tau) e^{j\Omega t}, \\ G_\tau(\omega) &= e^{-j(\omega - \Omega)\tau} P(\omega - \Omega), \end{aligned} \quad (11.3)$$

for  $\tau, \Omega \in \mathbb{R}$ , and then define a new transform by taking the inner product between  $x$  and  $g_{\Omega, \tau}(t)$  as

$$X(\Omega, \tau) = \langle x, g_{\Omega, \tau} \rangle = \int_{t \in \mathbb{R}} x(t) p(t - \tau) e^{-j\Omega t} dt, \quad (11.4)$$

that is, this new transform  $X(\Omega, \tau)$  is the Fourier transform of the windowed function  $x_\tau$  as in (11.2).





**Figure 11.1:** Local Fourier transform. The prototype function  $p(t)$  is centered at  $\tau$ , and thus, the Fourier transform only *sees* the neighborhood around  $\tau$ . For simplicity, a hat prototype function is shown; in practice, smoother ones are used.

From the construction, it is clear why this is called local Fourier transform, as shown in Figure 11.1. We are now ready to formally define it:

**DEFINITION 11.1 (LOCAL FOURIER TRANSFORM)** The local Fourier transform of a function  $x(t)$  is a function of  $\Omega, \tau \in \mathbb{R}$  given by

$$X(\Omega, \tau) = \langle x, g_{\Omega, \tau} \rangle = \int_{t \in \mathbb{R}} x(t) p(t - \tau) e^{-j\Omega t} dt, \quad \Omega, \tau \in \mathbb{R}. \quad (11.5a)$$

The inverse local Fourier transform of  $X(\Omega, \tau)$  is

$$x(t) = \frac{1}{2\pi} \int_{\Omega \in \mathbb{R}} \int_{\tau \in \mathbb{R}} X(\Omega, \tau) g_{\Omega, \tau}(t) d\Omega d\tau. \quad (11.5b)$$

To denote such a local Fourier-transform pair, we write:

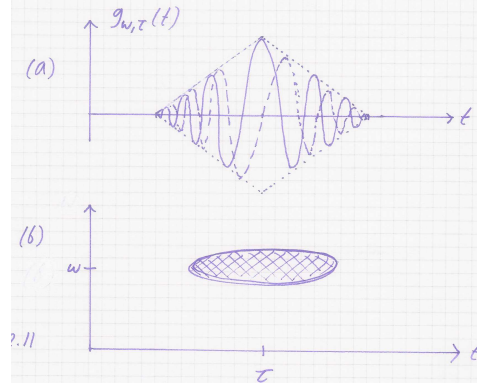
$$x(t) \xleftrightarrow{\text{LFT}} X(\Omega, \tau).$$

We will prove the inversion formula (11.5b) in a moment. For the analysis of a function  $x(t)$ , the  $X(\Omega, \tau)$  uses time-frequency atoms  $g_{\Omega, \tau}(t)$  that are centered around  $\Omega$  and  $\tau$ , as shown schematically in Figure 11.2. The local Fourier transform is highly redundant, mapping a one-dimensional function  $x(t)$  into a two-dimensional transform  $X(\Omega, \tau)$ .

**Prototype Window Function** The prototype function  $p(t)$  is critical in the local Fourier transform. The classical choice for  $p(t)$  is the unit-norm version of the Gaussian function given in (3.13a) with  $\gamma = (2\alpha/\pi)^{1/4}$ :

$$p(t) = \left(\frac{2\alpha}{\pi}\right)^{1/4} e^{-\alpha(t-\mu)^2}, \quad (11.6)$$

<sup>148</sup>Otherwise, it will interfere with the smoothness of the function to be analyzed; see Section 11.2.2.



**Figure 11.2:** Time-frequency atom used in the local Fourier transform. (a) Time-domain waveform  $g_{\omega, \tau}(t)$ . The prototype function is the hat function, and the real and imaginary parts of the complex exponential-modulated prototype function are shown. (b) Schematic time-frequency footprint of  $g_{\omega, \tau}(t)$ .

where  $\alpha$  is a scale parameter allowing us to tune the time resolution of the local Fourier transform.

Another classic choice is the unit-norm sinc function from (3.75),

$$p(t) = \sqrt{\frac{\omega_0}{2\pi}} \frac{\sin \omega_0 t/2}{\omega_0 t/2}, \quad (11.7)$$

that is, a perfect lowpass of bandwidth  $|\omega| \leq \omega_0/2$ . Here, the scale parameter allows us to tune the frequency resolution of the local Fourier transform.

Other prototype functions of choice include rectangular, triangular (hat) or higher-order spline functions, as well as other classic prototype functions from spectral analysis. An example is the *Hanning*, or, *raised cosine window* (we have seen its discrete counterpart in (2.15)), its unit-norm version defined as

$$p(t) = \begin{cases} \sqrt{\frac{2}{3\alpha}} (1 + \cos(2\pi t/\alpha)), & |t| \leq \alpha/2; \\ 0, & \text{otherwise,} \end{cases}$$

where  $\alpha$  is a scale parameter.<sup>149</sup>

**Inversion of the Local Fourier Transform** While we have taken for granted that the inversion formula (11.5b) holds, this is not a given. However, given the redundancy present in the local Fourier transform, we expect such an inversion to be possible, which we now prove.

We are going to apply the generalized Parseval's equality to (11.5b), and we

<sup>149</sup>In the signal processing literature, the normalization factor is usually  $1/2$ , such that  $p(0) = 1$ .

## 11.2. Local Fourier Transform

779

thus need the Fourier transform of  $X(\Omega, \tau)$  with respect to  $\tau$ . We have that

$$\begin{aligned} X(\Omega, \tau) &= \int_{t \in \mathbb{R}} x(t) p(t - \tau) e^{-j\Omega t} dt, \\ &\stackrel{(a)}{=} \int_{t \in \mathbb{R}} p(\tau - t) x(t) e^{-j\Omega t} dt \stackrel{(b)}{=} (p * x_\Omega)(\tau), \end{aligned}$$

where (a) follows from  $p(t) = p(-t)$ , and in (b) we introduced  $x_\Omega(t) = x(t)e^{-j\Omega t}$ . Using the shift-in-frequency property (3.57), the Fourier transform of  $x_\Omega(t)$  is  $X(\omega + \Omega)$ . Then, using the convolution property (3.64), the Fourier transform of  $X(\Omega, \tau)$  with respect to  $\tau$  becomes,

$$X(\Omega, \omega) = P(\omega) X(\omega + \Omega). \quad (11.8)$$

In (11.5b), the other term involving  $\tau$  is  $g_{\Omega, \tau}(t) = p(t - \tau)e^{j\Omega t}$ . Using the shift-in-time property (3.56) and because  $p(t)$  is symmetric, the Fourier transform of  $p(t - \tau)$  with respect to  $\tau$  is

$$p(t - \tau) \xrightarrow{\text{FT}} e^{-j\omega t} P(\omega). \quad (11.9)$$

We now apply the generalized Parseval's equality (3.69b) to the right side of (11.5b):

$$\begin{aligned} &\frac{1}{2\pi} \int_{\Omega \in \mathbb{R}} \left( \int_{\tau \in \mathbb{R}} X(\Omega, \tau) p(t - \tau) e^{j\Omega t} d\tau \right) d\Omega \\ &\stackrel{(a)}{=} \frac{1}{2\pi} \int_{\Omega \in \mathbb{R}} \left( \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} X(\omega + \Omega) P(\omega) P^*(\omega) e^{j\omega t} e^{j\Omega t} d\omega \right) d\Omega \\ &\stackrel{(b)}{=} \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} |P(\omega)|^2 \left( \frac{1}{2\pi} \int_{\Omega \in \mathbb{R}} X(\omega + \Omega) e^{j(\omega + \Omega)t} d\Omega \right) d\omega, \\ &\stackrel{(c)}{=} x(t) \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} |P(\omega)|^2 d\omega \stackrel{(d)}{=} x(t). \end{aligned}$$

where (a) follows from (11.8), (11.9) and generalized Parseval's equality (3.69b); (b) from Fubini's theorem (see Appendix 1.A.3) allowing for the exchange of the order of integration; (c) from the inverse Fourier transform (3.48b); and (d) from  $p$  being of unit norm and Parseval's equality (3.69a).

## 11.2.2 Properties of the Local Fourier Transform

We now look into the main properties of the local Fourier transform, including energy conservation, followed by basic characteristics such as localization properties and examples, including spectrograms, which are density plots of the magnitude of the local Fourier transform.

TBD: Table with properties.

**Linearity** The local Fourier transform operator is a linear operator, or,

$$\alpha x(t) + \beta y(t) \xrightarrow{\text{LFT}} \alpha X(\Omega, \tau) + \beta Y(\Omega, \tau). \quad (11.10)$$

**Shift in Time** A shift in time by  $t_0$  results in

$$x(t - t_0) \xleftrightarrow{\text{LFT}} e^{-j\Omega t_0} X(\Omega, \tau - t_0). \quad (11.11)$$

This is to be expected as it follows from the shift-in-time property of the Fourier transform, (3.56). To see that,

$$\begin{aligned} \int_{t \in \mathbb{R}} p(t - \tau) x(t - t_0) e^{-j\Omega t} dt &\stackrel{(a)}{=} e^{-j\Omega t_0} \int_{t' \in \mathbb{R}} p(t' - (\tau - t_0)) x(t') e^{-j\Omega t'} dt' \\ &\stackrel{(b)}{=} e^{-j\Omega t_0} X(\Omega, \tau - t_0), \end{aligned}$$

where (a) follows from change of variable  $t' = t - t_0$ ; and (b) from the definition of the local Fourier transform (11.5a). Thus, a shift by  $t_0$  simply shifts the local Fourier transform and adds a phase factor. The former illustrates the locality of the local Fourier transform, while the latter follows from the equivalent Fourier-transform property.

**Shift in Frequency** A shift in frequency by  $\omega_0$  results in

$$e^{j\omega_0 t} x(t) \xleftrightarrow{\text{LFT}} X(\Omega - \omega_0, \tau). \quad (11.12)$$

To see this,

$$\begin{aligned} &\int_{t \in \mathbb{R}} p(t - \tau) e^{j\omega_0 t} x(t) e^{-j\Omega t} dt \\ &= \int_{t \in \mathbb{R}} p(t - \tau) x(t) e^{-j(\Omega - \omega_0)t} dt = X(\Omega - \omega_0, \tau), \end{aligned}$$

the same as for the Fourier transform. As before, a shift in frequency is often referred to as *modulation*, and is dual to the shift in time.

**Parseval's Equality** The local Fourier-transform operator is a unitary operator and thus preserves the Euclidean norm (see (1.51)):

$$\|x\|^2 = \int_{t \in \mathbb{R}} |x(t)|^2 dt = \frac{1}{2\pi} \int_{\Omega \in \mathbb{R}} \int_{\tau \in \mathbb{R}} |X(\Omega, \tau)|^2 d\Omega d\tau = \frac{1}{2\pi} \|X\|^2. \quad (11.13)$$

We now prove Parseval's equality for functions that are both in  $\mathcal{L}^1$  and  $\mathcal{L}^2$ : It should come as no surprise that to derive Parseval's equality for the local Fourier transform, we use Parseval's equality for the Fourier transform. Start with the right side of (11.13) to get

$$\begin{aligned} &\frac{1}{2\pi} \int_{\tau \in \mathbb{R}} \int_{\Omega \in \mathbb{R}} |X(\Omega, \tau)|^2 d\Omega d\tau \\ &\stackrel{(a)}{=} \frac{1}{2\pi} \int_{\Omega \in \mathbb{R}} \left( \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} |X(\omega + \Omega) P(\omega)|^2 d\omega \right) d\Omega \\ &\stackrel{(b)}{=} \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} |P(\omega)|^2 \left( \frac{1}{2\pi} \int_{\Omega \in \mathbb{R}} |X(\omega + \Omega)|^2 d\Omega \right) d\omega, \\ &\stackrel{(c)}{=} \|x\|^2 \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} |P(\omega)|^2 d\omega \stackrel{(d)}{=} \|x\|^2, \end{aligned}$$

## 11.2. Local Fourier Transform

781

where (a) follows from Parseval's equality (3.69a) and (11.8); (b) from Fubini's theorem (see Appendix 1.A.3) allowing for the exchange of the order of integration; (c) from the inverse Fourier transform (3.48b); and (d) from  $p$  being of unit norm and Parseval's equality (3.69a).

**Redundancy** The local Fourier transform maps a function of one variable into a function of two variables. It is thus highly redundant, and this redundancy is expressed by the *reproducing kernel*:

$$\begin{aligned} K(\Omega, \tau, \omega_0, t_0) &= \langle g_{\Omega, \tau}, g_{\omega_0, t_0} \rangle \\ &= \int_{t \in \mathbb{R}} p(t - \tau) p(t - t_0) e^{j(\omega_0 - \Omega)t} dt. \end{aligned} \quad (11.14)$$

While this is a four-dimensional object, its magnitude depends only on the two differences  $(\omega_0 - \Omega)$  and  $(t_0 - \tau)$ .<sup>150</sup>

**PROPOSITION 11.2 (REPRODUCING KERNEL FORMULA FOR THE LOCAL FOURIER TRANSFORM)**  
A function  $X(\omega_0, t_0)$  is the local Fourier transform of some function  $x(t)$  if and only if it satisfies

$$X(\Omega, \tau) = \frac{1}{2\pi} \int_{t_1 \in \mathbb{R}} \int_{\omega_0 \in \mathbb{R}} X(\omega_0, t_0) K(\Omega, \tau, \omega_0, t_0) d\omega_0 dt_0. \quad (11.15)$$

*Proof.* If  $X(\omega_0, t_0)$  is a local Fourier transform, then there is a function  $x(t)$  such that  $X(\omega_0, t_0) = X(\omega_0, t_0)$ , or

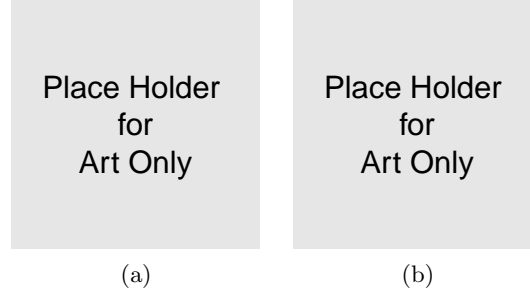
$$\begin{aligned} X(\Omega, \tau) &= \int_{t \in \mathbb{R}} x(t) g_{\Omega, \tau}^*(t) dt, \\ &\stackrel{(a)}{=} \int_{t \in \mathbb{R}} \left( \frac{1}{2\pi} \int_{t_0 \in \mathbb{R}} \int_{\omega_0 \in \mathbb{R}} X(\omega_0, t_0) g_{\omega_0, t_0}(t) d\omega_0 dt_0 \right) g_{\Omega, \tau}^*(t) dt \\ &= \frac{1}{2\pi} \int_{t_0 \in \mathbb{R}} \int_{\omega_0 \in \mathbb{R}} X(\omega_0, t_0) \left( \int_{t \in \mathbb{R}} g_{\Omega, \tau}(t) g_{\omega_0, t_0}(t) dt \right) d\omega_0 dt_0, \\ &\stackrel{(b)}{=} \frac{1}{2\pi} \int_{t_0 \in \mathbb{R}} \int_{\omega_0 \in \mathbb{R}} X(\omega_0, t_0) K(\Omega, \tau, \omega_0, t_0) d\omega_0 dt_0, \end{aligned}$$

(11.5b) where (a) follows from the inversion formula (11.5b) and (b) from (11.15).

For the converse, write (11.15) by making  $K(\Omega, \tau, \omega_0, t_0)$  explicit as an integral over  $t$  (see (11.14)):

$$\begin{aligned} X(\Omega, \tau) &= \frac{1}{2\pi} \int_{t_1 \in \mathbb{R}} \int_{\omega_1 \in \mathbb{R}} \int_{t \in \mathbb{R}} X(\omega_0, t_0) g_{\Omega, \tau}(t) g_{\omega, t}(t) dt d\omega_0 dt_0 \\ &\stackrel{(a)}{=} \int_{t \in \mathbb{R}} g_{\Omega, \tau}(t) \left( \frac{1}{2\pi} \int_{t_1 \in \mathbb{R}} \int_{\omega_1 \in \mathbb{R}} X(\omega_0, t_0) g_{\omega, t}(t) d\omega_0 dt_0 \right) dt \\ &\stackrel{(b)}{=} \int_{t \in \mathbb{R}} g_{\Omega, \tau}(t) x(t) dt, \end{aligned}$$

<sup>150</sup>This is expressed in a closely related function called the *ambiguity function*; see Exercise 11.1.



**Figure 11.3:** Localization properties of the local Fourier transform. (a) A function perfectly localized in time, a Dirac delta function at  $\tau$ , with a compactly supported prototype function  $[-T/2, T/2]$ . (b) A function perfectly localized in frequency, a complex exponential function of frequency  $\omega$ , with a prototype function having a compactly supported Fourier transform  $[-B/2, B/2]$ .

where (a) follows from Fubini's theorem (see Appendix 1.A.3) allowing for the exchange of the order of integration, and (b) from the inversion formula (11.5b). Therefore,  $X(\Omega, \tau)$  is indeed a local Fourier transform, namely the local Fourier transform of  $x(t)$ .

The redundancy present in the local Fourier transform allows sampling and interpolation, and the interpolation kernel depends on the reproducing kernel.

**Characterization of Singularities and Smoothness** To characterize singularities, we will take the view that the local Fourier transform is a Fourier transform of a windowed function  $x_\tau(t)$  as in (11.1). Since this is a product between the function and the prototype function, using the convolution-in-frequency property (3.65), in the Fourier domain this is a convolution. That is, singularities are smoothed by the prototype function.

We now characterize singularities in time and frequency, depicted in Figure 11.3.

- (i) *Characterization of singularities in time:* Take a function perfectly localized in time, the Dirac delta function  $x(t) = \delta(t - t_0)$ . Then

$$X(\Omega, \tau) = \int_{t \in \mathbb{R}} p(t - \tau) \delta(t - t_0) e^{-j\Omega t} dt \stackrel{(a)}{=} p(t_0 - \tau) e^{-j\Omega t_0},$$

where (a) follows from Table 3.1. This illustrates the characterization of singularities in time by the local Fourier transform: An event at time location  $t_0$  will spread around  $t_0$  according to the prototype function, and this across all frequencies. If  $p(t)$  has compact support  $[-T/2, T/2]$ , then  $X(\Omega, \tau)$  has support  $[-\infty, \infty] \times [t_0 - T/2, t_0 + T/2]$ .

- (ii) *Characterization of singularities in frequency:* Take now a function perfectly

localized in frequency, a complex exponential function  $x(t) = e^{j\omega t}$ . Then,

$$\begin{aligned} X(\Omega, \tau) &= \int_{t \in \mathbb{R}} p(t - \tau) e^{-j(\Omega - \omega)t} dt \\ &\stackrel{(a)}{=} e^{-j(\Omega - \omega)\tau} \int_{t' \in \mathbb{R}} p(t') e^{-j(\Omega - \omega)t'} dt' \\ &\stackrel{(b)}{=} e^{-j(\Omega - \omega)\tau} P(\Omega - \omega), \end{aligned} \quad (11.16)$$

where (a) follows from change of variables  $t' = t - \tau$ ; and (b) from the Fourier transform of  $p(t)$ . This illustrates the characterization of singularities in frequency by the local Fourier transform: An event at frequency location  $\omega$  will spread around  $\omega$  according to the prototype function, and this across all time. If  $P(\omega)$  has compact support  $[-B/2, B/2]$ , then  $X(\Omega, \tau)$  has support  $[\omega - B/2, \omega + B/2] \times [-\infty, \infty]$ .

What is important to understand is that if singularities appear together within a prototype function, they appear mixed in the local Fourier transform domain. This is unlike the continuous wavelet transform we will see in the next chapter, where arbitrary time resolution is possible for the scale factor going to 0.

If the prototype function is smoother than the function to be analyzed, then the type of singularity (assuming there is a single one inside the prototype function) is determined by the decay of the Fourier transform.

**EXAMPLE 11.1 (SINGULARITY CHARACTERIZATION OF THE LOCAL FOURIER TRANSFORM)**  
Let us consider, as an illustrative example, a hat prototype function from (3.49a):

$$p(t) = \begin{cases} 1 - |t|, & |t| < 1; \\ 0, & \text{otherwise,} \end{cases}$$

which has a Fourier transform (3.49f) decaying as  $|\omega|^{-2}$  for large  $\omega$ .

Consider a function  $x(t) \in C^1$  (continuous and with at least one continuous derivative, see Section 1.2.4) except for a discontinuity at  $t = t_0$ . If it were not for the discontinuity, the Fourier transform of  $x(t)$  would decay faster than  $|\omega|^{-2}$  (that is, faster than  $|P(\omega)|$  does). However, because of the singularity at  $t = t_0$ ,  $|X(\omega)|$  decays only as  $|\omega|^{-1}$ .

Now the locality of the local Fourier transform comes into play. There are two modes, given by the regularity of the windowed function  $x_\tau(t)$ : (1) When  $\tau$  is far from  $t_0$ ,  $|\tau - t_0| > 1$ ,  $x_\tau(t)$  is continuous (but its derivative is not, because of the hat prototype function), and  $|X(\Omega, \tau)|$  decays as  $|\omega|^{-2}$ . (2) When  $\tau$  is close to  $t_0$ ,  $|\tau - t_0| \leq 1$ , that is, it is close to the discontinuity,  $x_\tau(t)$  is discontinuous, and  $|X(\Omega, \tau)|$  decays only as  $|\omega|^{-1}$ .

This above example indicates that there is a subtle interplay between the smoothness and support of the prototype function, and the singularities or smoothness of the analyzed function. This is formalized in the following two results:

**PROPOSITION 11.3 (SINGULARITY CHARACTERIZATION OF THE LOCAL FOURIER TRANSFORM)**

Assume a prototype function  $p(t)$  with compact support  $[-T/2, T/2]$  and sufficient smoothness. Consider a function  $x(t)$  which is smooth except for a singularity of order  $n$  at  $t = t_0$ , that is, its  $n$ th derivative at  $t_0$  is a Dirac delta function. Then its local Fourier transform decays as

$$|X(\Omega, \tau)| \sim O\left(\frac{1}{1 + |\omega|^n}\right)$$

in the region  $\tau \in [t_0 - T/2, t_0 + T/2]$ .

The proof follows by using the decay property of the Fourier transform applied to the windowed function and is left as Exercise 11.2.

Conversely, a sufficiently decaying local Fourier transform indicates a smooth function in the region of interest.

**PROPOSITION 11.4 (SMOOTHNESS FROM DECAY OF THE LOCAL FOURIER TRANSFORM)**

Consider a sufficiently smooth prototype function  $p(t)$  of compact support  $[-T/2, T/2]$ . If the local Fourier transform at  $t_0$  decays sufficiently fast, or for some  $\alpha$  and  $\epsilon > 0$ ,

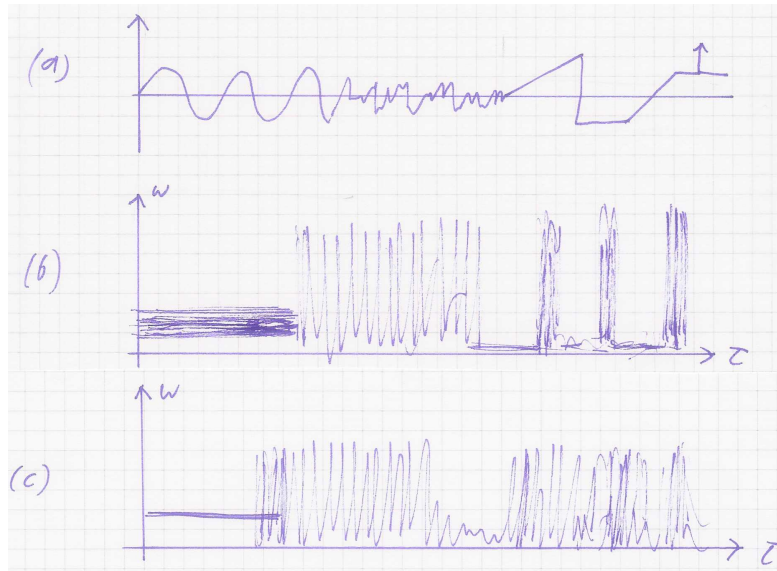
$$|X(\Omega, \tau)| \leq \frac{\alpha}{1 + |\Omega|^{p+1+\epsilon}}$$

then  $x(t)$  is  $C^p$  on the interval  $[t_0 - T/2, t_0 + T/2]$ .

**Spectrograms** The standard way to display the local Fourier transform is as a density plot of  $|X(\Omega, \tau)|$ . This is called the *spectrogram* and is very popular, for example, for speech and music signals. Figure 11.4 shows a standard signal with various modes and two spectrograms.

As can be seen, the sinusoid is chosen, and the singularities are identified but not exactly localized due to the size of the prototype function. For the short prototype function in (Figure 11.4(b)), various singularities are still isolated, but the sinusoid is not well localized. The reverse is true for the long prototype function (Figure 11.4(c)), where the sinusoid is well identified, but some of the singularities are now mixed together. This is of course the fundamental tension between time and frequency localization, as governed by the uncertainty principle we have seen in Chapter 6.





**Figure 11.4:** The spectrogram. (a) A signal with various modes. (b) The spectrogram, or  $|X(\Omega, \tau)|$ , with a short prototype function. (c) The spectrogram with a long prototype function.

## 11.3 Local Fourier Frame Series

### 11.3.1 Sampling Grids

### 11.3.2 Frames from Sampled Local Fourier Transform

## 11.4 Local Fourier Series

### 11.4.1 Complex Exponential-Modulated Local Fourier Bases

#### Complex-Exponential Modulation

#### Balian-Low Theorem

### 11.4.2 Cosine-Modulated Local Fourier Bases

#### Cosine Modulation

**Local Cosine Bases****11.5 Computational Aspects****11.5.1 Complex Exponential-Modulated Local Fourier Bases****11.5.2 Cosine-Modulated Local Fourier Bases****Chapter at a Glance**

TBD

**Historical Remarks**

TBD

**Further Reading**

TBD

---

**Exercises with Solutions**

11.1. TBD

---

**Exercises**11.1. *Ambiguity Function and Reproducing Kernel of the Local Fourier Transform*11.2. *Smoothness and Decay of the Local Fourier Transform*

Prove Propositions 11.3 and 11.4.

*(Hint: Use appropriate Fourier-transform relations. Make sure the smoothness of the window function is properly taken into account.)*

## Chapter 12

# Wavelet Bases, Frames and Transforms on Functions

The previous chapter started with the most redundant version of the local Fourier expansions on functions: the local Fourier transform. We lowered its redundancy through sampling, leading to Fourier frames. Ultimately, we wanted to make it nonredundant by trying to build local Fourier bases; however, we hit a roadblock, the Balian-Low theorem, prohibiting such bases with reasonable joint time and frequency localization. While bases are possible with cosine, instead of complex-exponential, modulation, we can do even better. In this chapter, we will start by constructing wavelet bases, and then go in the direction of increasing redundancy, by building frames and finally the continuous wavelet transform.

## 12.1 Introduction

Iterated filter banks from Chapter 9 pose interesting theoretical and practical questions, the key one quite simple: what happens if we iterate the DWT to infinity? While we need to make the question precise by indicating how this iterative process takes place, when done properly, and under certain conditions on the filters used in the filter bank, the limit leads to a wavelet basis for the space of square-integrable functions,  $\mathcal{L}^2(\mathbb{R})$ . The key notion is that we take a discrete-time basis (orthonormal or biorthogonal) for  $\ell^2(\mathbb{Z})$ , and derive from it a continuous-time basis for  $\mathcal{L}^2(\mathbb{R})$ . This connection between discrete and continuous time is reminiscent of the concepts and aim of Chapter 4, including the sampling theorem. The iterative process itself is fascinating, but the resulting bases are even more so: they are scale invariant (as opposed to shift invariant) so that all basis vectors are obtained from a single function  $\psi(t)$  through shifting and scaling. What we do in this opening section is go through some salient points on a simple example we have seen numerous times in Part II: the Haar basis. We will start from its discrete-time version seen in Chapter 7 (level 1, scale 0) and the iterated one seen in Chapter 9 (level  $J$ , scale  $2^J$ ) and build a continuous-time basis for  $\mathcal{L}^2(\mathbb{R})$ . We then mimic this process and show how it can lead to a wealth of different wavelet bases. We will also look into the Haar frame and Haar continuous wavelet transform. We follow the same roadmap

from this section, iterated filters—wavelet series—wavelet frame series—continuous wavelet transform, throughout the rest of the chapter, but in a more general setting. As the chapter contains a fair amount of material, some of it quite technical, this section attempts to cover all the main concepts, and is thus rather long. The details in more general settings are covered throughout the rest of the chapter.

### 12.1.1 Scaling Function and Wavelets from Haar Filter Bank

To set the stage, we start with the Haar filters  $g$  and  $h$  given in Table 7.8, Chapter 7, where we used their impulse responses and shifts by multiples by two as a basis for  $\ell^2(\mathbb{Z})$ . This orthonormal basis was implemented using a critically-sampled two-channel filter bank with down- and upsampling by 2, synthesis lowpass/highpass filter pair  $g_n, h_n$  from (7.1) (repeated here for easy reference)

$$g_n = \frac{1}{\sqrt{2}}(\delta_n + \delta_{n-1}) \xleftrightarrow{\text{ZT}} G(z) = \frac{1}{\sqrt{2}}(1 + z^{-1}) \quad (12.1a)$$

$$h_n = \frac{1}{\sqrt{2}}(\delta_n - \delta_{n-1}) \xleftrightarrow{\text{ZT}} H(z) = \frac{1}{\sqrt{2}}(1 - z^{-1}), \quad (12.1b)$$

and a corresponding analysis lowpass/highpass filter pair  $g_{-n}, h_{-n}$ .

We then used these filters and the associated two-channel filter bank as a building block for the Haar DWT in Chapter 9. For example, we saw that in a 3-level iterated Haar filter bank, the lowpass and highpass at level 3 were given by (9.1c)–(9.1d) and plotted in Figure 9.4:

$$\begin{aligned} G^{(3)}(z) &= G(z)G(z^2)G(z^4) \\ &= \frac{1}{2\sqrt{2}}(1 + z^{-1})(1 + z^{-2})(1 + z^{-4}) \\ &= \frac{1}{2\sqrt{2}}(1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + z^{-5} + z^{-6} + z^{-7}), \end{aligned} \quad (12.2a)$$

$$\begin{aligned} H^{(3)}(z) &= G(z)G(z^2)H(z^4) \\ &= \frac{1}{2\sqrt{2}}(1 + z^{-1})(1 + z^{-2})(1 - z^{-4}) \\ &= \frac{1}{2\sqrt{2}}(1 + z^{-1} + z^{-2} + z^{-3} - z^{-4} - z^{-5} - z^{-6} - z^{-7}). \end{aligned} \quad (12.2b)$$

**Iterated Filters** We now revisit these filters and their iterations, but with a new angle, as we let the iteration go to infinity by associating a continuous-time function to the discrete-time sequence (impulse response of the iterated filter).

We first write the expressions for the equivalent filters at the last level of a

$J$ -level iterated Haar filter bank:

$$\begin{aligned}
 G^{(J)}(z) &= \prod_{\ell=0}^{J-1} G(z^{2^\ell}) = 2^{-J/2} \prod_{\ell=0}^{J-1} (1 + z^{-2^\ell}) \\
 &= 2^{-J/2} \sum_{n=0}^{2^J-1} z^{-n} = G^{(J-1)}(z) \underbrace{\frac{1}{\sqrt{2}}(1 + z^{2^{J-1}})}_{G(z^{2^{J-1}})}, \\
 H^{(J)}(z) &= \prod_{\ell=0}^{J-2} G(z^{2^\ell}) H(z^{2^{J-1}}) = 2^{-J/2} \prod_{\ell=0}^{J-2} (1 + z^{-2^\ell})(1 - z^{-2^{J-1}}) \\
 &= 2^{-J/2} \left( \sum_{n=0}^{2^{J-1}-1} z^{-n} - \sum_{n=2^{J-1}}^{2^J-1} z^{-n} \right) = G^{(J-1)}(z) \underbrace{\frac{1}{\sqrt{2}}(1 - z^{2^{J-1}})}_{H(z^{2^{J-1}})}.
 \end{aligned} \tag{12.3}$$

We have seen the above expressions in (9.5c), (9.9) already; they construct the equivalent filter at the subsequent level, from the equivalent filters at the previous one.

We know that, by construction, these filters are orthonormal to their shifts by  $2^J$ , (9.6a), (9.11a), as well as orthogonal to each other, (9.14a), and their lengths are

$$L^{(J)} = 2^J. \tag{12.4}$$

**Scaling Function and its Properties** We now associate a piecewise-constant function  $\varphi^{(J)}(t)$  to  $g_n^{(J)}$  so that  $\varphi^{(J)}(t)$  is of finite length and norm 1; we thus have to determine the width and height of the piecewise segments. Since the number of piecewise segments (equal to the number of nonzero coefficients of  $g_n^{(J)}$ ) grows exponentially with  $J$  because of (12.4), we choose their width as  $2^{-J}$ , upper bounding the length of  $\varphi^{(J)}(t)$  by 1. For  $\varphi^{(J)}(t)$  to inherit the unit-norm property from  $g_n^{(J)}$ , we choose the height of the piecewise segments as  $2^{J/2}g_n^{(J)}$ . Then, the  $n$ th piece of  $\varphi^{(J)}(t)$  contributes

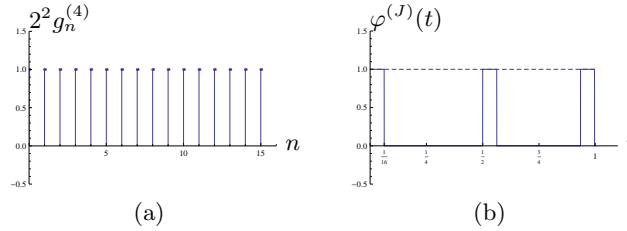
$$\int_{n/2^J}^{(n+1)/2^J} |\varphi^{(J)}(t)|^2 dt = \int_{n/2^J}^{(n+1)/2^J} 2^J (g_n^{(J)})^2 dt = (g_n^{(J)})^2 \stackrel{(a)}{=} 2^{-J}$$

to  $\varphi^{(J)}(t)$  (where (a) follows from (12.3)). Summing up the individual contributions,

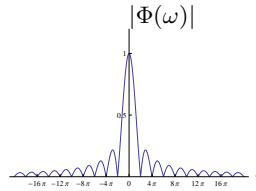
$$\|\varphi^{(J)}(t)\|^2 = \sum_{n=0}^{2^J-1} \int_{n/2^J}^{(n+1)/2^J} |\varphi^{(J)}(t)|^2 dt = \sum_{n=0}^{2^J-1} 2^{-J} = 1$$

as in Figure 12.1. We have thus defined our piecewise-constant function as

$$\varphi^{(J)}(t) = 2^{J/2} g_n^{(J)} = 1 \quad \frac{n}{2^J} \leq t < \frac{n+1}{2^J}. \tag{12.5}$$



**Figure 12.1:** Example construction of a piecewise-constant function  $\varphi^{(J)}(t)$  from  $g_n^{(J)}$ . (a) Discrete-time sequence  $2^2 g_n^{(4)}$ . (b) Continuous-time piecewise-constant function  $\varphi^{(4)}(t)$  (we plot a few isolated piecewise segments for emphasis).



**Figure 12.2:** Magnitude response  $\Phi(\omega)$  of the scaling function  $\varphi(t)$ .

As  $\varphi^{(J)}(t)$  is 1 on every interval of length  $2^{-J}$  and  $g^{(J)}$  has exactly  $2^J$  nonzero entries (see (12.4)), this function is actually 1 on the interval  $[0, 1)$  for every  $J$ , that is, the limit of  $\varphi^{(J)}(t)$  is the indicator function of the unit interval  $[0, 1]$  (or, a box function shifted to  $1/2$ ),  $\varphi(t)$ , independently of  $J$ ,

$$\varphi^{(J)}(t) = \begin{cases} 1, & 0 \leq t < 1; \\ 0, & \text{otherwise,} \end{cases} = \varphi(t). \quad (12.6)$$

Convergence is achieved without any problem, actually in one step! <sup>151</sup>

The function  $\varphi(t)$  is called the *Haar scaling function*. Had we started with a different lowpass filter  $g$ , the resulting limit, had it existed, would have lead to a different scaling function, a topic we will address later in the chapter.

In the Fourier domain,  $\Phi^{(J)}(\omega)$ , the Fourier transform of  $\varphi^{(J)}(t)$ , will be the same for every  $J$  because of (12.6), and thus, the Fourier transform of the scaling function will be the sinc function in frequency (see Table 3.6 and Figure 12.2):

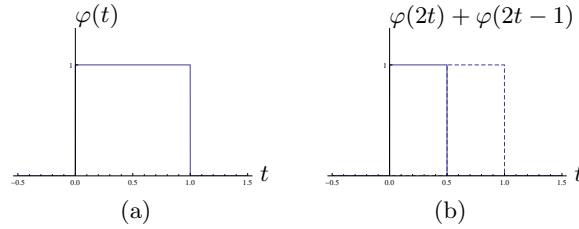
$$\Phi(\omega) = e^{-j\omega/2} \frac{\sin(\omega/2)}{\omega/2} = e^{-j\omega/2} \text{sinc}(\omega/2). \quad (12.7)$$

We now turn our attention to some interesting properties of the scaling function:

(i) *Two-scale equation:* The Haar scaling function  $\varphi(t)$  satisfies

$$\varphi(t) = \sqrt{2}(g_0 \varphi(2t) + g_1 \varphi(2t - 1)) = \varphi(2t) + \varphi(2t - 1), \quad (12.8)$$

<sup>151</sup>Although we could have defined a piecewise-linear function instead of a piecewise-constant one, we chose not to do so as the behavior of the limit we will study does not change.



**Figure 12.3:** Two-scale equation for the Haar scaling function. (a) The scaling function  $\varphi(t)$  and (b) expressed as a linear combination of  $\varphi(2t)$  and  $\varphi(2t-1)$ .

the so-called two-scale equation. We see that the scaling function is built out of two scaled versions of itself, illustrated in Figure 12.3. While in this Haar case, this does not come as a big surprise, it will when the scaling functions become more complex. To find the expression for the two-scale equation in the Fourier domain, we rewrite (12.8) as a convolution

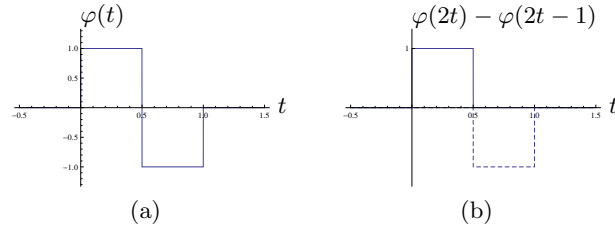
$$\varphi(t) = \varphi(2t) + \varphi(2t-1) = \sqrt{2} \sum_{k=0}^1 g_k \varphi(2t-k). \quad (12.9)$$

We can then use the convolution-in-time property (3.64) and the scaling property (3.58a) of the Fourier transform to get

$$\Phi(\omega) = \frac{1}{\sqrt{2}} G(e^{j\omega/2}) \Phi(\omega/2) = \frac{1}{2} (1 + e^{-j\omega/2}) e^{-j\omega/4} \text{sinc}(\omega/4). \quad (12.10)$$

- (ii) *Smoothness:* The Haar scaling function  $\varphi(t)$  is not continuous.<sup>152</sup> This can also be seen from the decay of its Fourier transform  $\Phi(\omega)$ , which, as we know from (12.7), is a sinc function (see also Figure 12.2), and thus decays slowly (it has, however, only two points of discontinuity and is, therefore, not altogether ill-behaved).
- (iii) *Reproduction of polynomials:* The Haar scaling function  $\varphi(t)$  with its integer shifts can reproduce constant functions. This stems from the polynomial approximation properties of  $g$ , as in Theorem 7.5. In the next section, we will see how other scaling functions will be able to reproduce polynomials of degree  $N$ . The key will be the number of zeros at  $\omega = \pi$  of the lowpass filter  $G(e^{j\omega})$ ; from (12.1a), for the Haar scaling function, there is just 1.
- (iv) *Orthogonality to integer shifts:* The Haar scaling function  $\varphi(t)$  is orthogonal to its integer shifts, another property inherited from the underlying filter. Of course, in this Haar case the property is obvious, as the support of  $\varphi(t)$  is limited to the unit interval. The property will still hold for more general scaling functions, albeit it will not be that obvious to see.

<sup>152</sup>In Proposition 12.1, we will see a sufficient condition for the limit function, if it exists, to be continuous (and possibly  $k$ -times differentiable).



**Figure 12.4:** Two-scale equation for the Haar wavelet. (a) The wavelet  $\psi(t)$  and (b) expressed as a linear combination of  $\varphi(2t)$  and  $\varphi(2t - 1)$ .

**Wavelet and its Properties** The scaling function we have just seen is lowpass in nature (if the underlying filter  $g$  is lowpass in nature). Similarly, we can construct a *wavelet* (or, simply *wavelet*) that will be bandpass in nature (if the underlying filter  $h$  is highpass in nature).

We thus associate a piecewise-constant function  $\psi^{(J)}(t)$  to  $h_n^{(J)}$  in such a way that  $\psi^{(J)}(t)$  is of finite length and of norm 1; we use the same arguments as before to determine the width and height of the piecewise segments, leading to

$$\psi^{(J)}(t) = 2^{J/2} h_n^{(J)} \quad \frac{n}{2^J} \leq t < \frac{n+1}{2^J}. \quad (12.11)$$

Like  $\varphi^{(J)}(t)$ , the function  $\psi^{(J)}(t)$  is again the same for every  $J$  since the length of  $h^{(J)}$  is exactly  $2^J$ . It is 1 for  $n = 0, 1, \dots, 2^{J-1} - 1$ , and is  $-1$  for  $n = 2^{J-1}, 2^{J-1} + 1, \dots, 2^J - 1$ . Thus, it comes as no surprise that the limit is

$$\psi(t) = \begin{cases} 1, & 0 \leq t < 1/2; \\ -1, & 1/2 \leq t < 1; \\ 0, & \text{otherwise,} \end{cases} \quad (12.12)$$

called the *Haar wavelet*, or, *Haar wavelet*,<sup>153</sup> (see Figure 12.4(a)).

Similarly to  $\Phi(\omega)$ , in the Fourier-domain,

$$\Psi(\omega) = \frac{1}{2}(1 - e^{-j\omega/2})e^{-j\omega/4} \operatorname{sinc}(\omega/4) = \frac{1}{\sqrt{2}} H(e^{j\omega/2}) \Phi(\omega/2). \quad (12.13)$$

We now turn our attention to some interesting properties of the Haar wavelet:

- (i) *Two-scale equation:* We can see from  $\Psi(\omega)$  its highpass nature as it is 0 at  $\omega = 0$  (because  $H(1) = 0$ ). Using the convolution-in-time property (3.64) and the scaling property (3.58a) of the Fourier transform, we see that the above can be written in the time domain as (see Figure 12.4(b))

$$\psi(t) = \sqrt{2} \langle h_k, \varphi(2t - k) \rangle_k = \varphi(2t) - \varphi(2t - 1). \quad (12.14)$$

In other words, the wavelet is built out of the scaling function at a different scale and its shift, its own two-scale equation, but involving scaled versions

<sup>153</sup>Depending on the initial discrete-time filters, the resulting limits, when they exist, lead to different wavelets.



of the scaling function instead of itself. The last expression in (12.13) is the Fourier-domain version of the two-scale equation.

- (ii) *Smoothness*: Since the wavelet is a linear combination of the scaled scaling function and its shift, its smoothness is inherited from the scaling function; in other words, like the scaling function, it is not continuous, having 3 points of discontinuity.
- (iii) *Zero-moment property*: We have seen that the Haar scaling function can reproduce constant functions. The Haar wavelet has a complementary property, called zero-moment property. To see that,

$$\Phi(\omega)|_{\omega=0} = \int_{-\infty}^{\infty} \psi(t) dt = 0,$$

that is, the inner product between the wavelet and a constant function will be zero. In other words, the wavelet annihilates constant functions while the scaling function reproduces them.

- (iv) *Orthogonality to integer shifts*: Finally, like the scaling function, the wavelet is orthogonal with respect to integer shifts. Again, this is trivial to see for the Haar wavelet as it is supported on the unit interval only.
- (v) *Orthogonality of the scaling and wavelets*: It is also trivial to see that the scaling function and the wavelet are orthogonal to each other. All these properties are setting the stage for us to build a basis based on these functions.

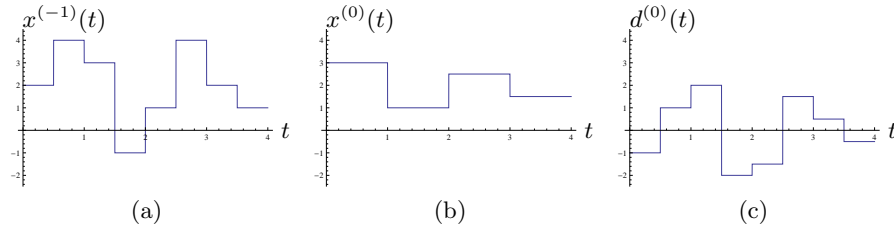
### 12.1.2 Haar Wavelet Series

Thus far, we have constructed two functions, the scaling function  $\varphi(t)$  and the wavelet  $\psi(t)$ , by iterating the Haar filter bank. That filter bank implements a discrete-time Haar basis, what about in continuous time? What we can say is that this scaling function and the wavelet, together with their integer shifts,  $\{\varphi(t - k), \psi(t - k)\}_{k \in \mathbb{Z}}$  do constitute a basis, for the space of piecewise-constant functions on intervals of half-integer length or more (see Figure 12.5(a)–(c)). We can see that as follows. Assume we are given a function  $x^{(-1)}(t)$  that equals  $a$  for  $0 \leq t < 1/2$ ;  $b$  for  $1/2 \leq t < 1$ ; and 0 otherwise. Then,

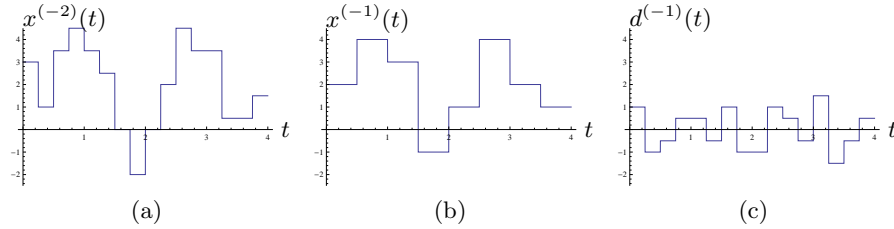
$$\begin{aligned} x^{(-1)}(t) &= \frac{a+b}{2} \varphi(t) + \frac{a-b}{2} \psi(t) \\ &= \langle x^{(-1)}(t), \varphi(t) \rangle \varphi(t) + \langle x^{(-1)}(t), \psi(t) \rangle \psi(t) \\ &= x^{(0)}(t) + d^{(0)}(t). \end{aligned} \tag{12.15}$$

Had the function  $x^{(-1)}(t)$  been nonzero on any other interval, we could have used the integer shifts of  $\varphi(t)$  and  $\psi(t)$ .

Clearly, this process scales by 2. In other words, the scaled scaling function and the wavelet, together with their shifts by multiples of  $1/2$ ,  $\{\sqrt{2}\varphi(2t - k), \sqrt{2}\psi(2t - k)\}_{k \in \mathbb{Z}}$  do constitute a basis, for the space of piecewise-constant functions on intervals of quarter-integer length or more (see Figure 12.6(a)–(c)). Assume, for



**Figure 12.5:** Haar series decomposition of (a)  $x^{(-1)}(t)$ , a function constant on half-integer intervals, using  $\{\varphi(t-k), \psi(t-k)\}_{k \in \mathbb{Z}}$ , into (b)  $x^{(0)}(t)$  and (c)  $d^{(0)}(t)$ .



**Figure 12.6:** Haar series decomposition of (a)  $x^{(-2)}(t)$ , a function constant on quarter-integer intervals, using  $\{\sqrt{2}\varphi(2t-k), \sqrt{2}\psi(2t-k)\}_{k \in \mathbb{Z}}$ , into (b)  $x^{(-1)}(t)$  and (c)  $d^{(-1)}(t)$ .

example, we are now given a function  $x^{(-2)}(t)$  that equals  $c$  for  $0 \leq t < 1/4$ ;  $d$  for  $1/4 \leq t < 1/2$ ; and 0 otherwise. Then,

$$\begin{aligned}
 x^{(-1)}(t) &= \frac{c+d}{2} \varphi(2t) + \frac{c-d}{2} \psi(2t) \\
 &= \langle x^{(-2)}, \sqrt{2}\varphi(2t) \rangle \sqrt{2}\varphi(2t) + \langle x^{(-2)}, \sqrt{2}\psi(2t) \rangle \sqrt{2}\psi(2t) \\
 &= x^{(-1)}(t) + d^{(-1)}(t) \stackrel{(a)}{=} x^{(0)}(t) + d^{(0)}(t) + d^{(-1)}(t),
 \end{aligned}$$

where (a) follows from (12.15). In other words, we could have also decomposed  $x^{(-2)}(t)$  using  $\{\varphi(t-k), \psi(t-k)\}_{k \in \mathbb{Z}}$ .

Continuing this argument, to represent piecewise-constant functions on inter-

## 12.1. Introduction

795

vals of length  $2^{-\ell}$ , we need the following basis:

Scale	Functions	
0	scaling function and its shifts	$\varphi(t - k)$
	wavelet and its shifts	$\psi(t - k)$
-1	wavelet and its shifts	$2^{1/2}\psi(2t - k)$
$\vdots$	$\vdots$	$\vdots$
$-(\ell - 1)$	wavelet and its shifts	$2^{-(\ell-1)/2}\psi(2^{-(\ell-1)}t - k)$

**Definition of the Haar Wavelet Series** From what we have seen, if we want to represent shorter and shorter constant pieces, we need to keep on adding wavelets with decreasing scale together with the scaling function at the coarsest scale. We may imagine, and we will formalize this in a moment, that if we let this process go to infinity, the scaling function will eventually become superfluous, and it does. This previous discussion leads to the Haar orthonormal set and a truly surprising result dating back to Haar in 1910, that this orthonormal system is in fact a basis for  $\mathcal{L}^2(\mathbb{R})$ . This result is the Haar continuous-time counterpart of Theorem 9.2, which states that the discrete-time wavelet  $h_n$  and its shifts and scales (equivalent iterated filters) form an orthonormal basis for the space of finite-energy sequences,  $\ell^2(\mathbb{Z})$ . The general result will be given by Theorem 12.6.

For compactness, we start by renaming our basis functions as:

$$\psi_{\ell,k}(t) = 2^{-\ell/2} \psi(2^{-\ell}t - k) = \frac{1}{2^{\ell/2}} \psi\left(\frac{t - 2^\ell k}{2^\ell}\right), \quad (12.16a)$$

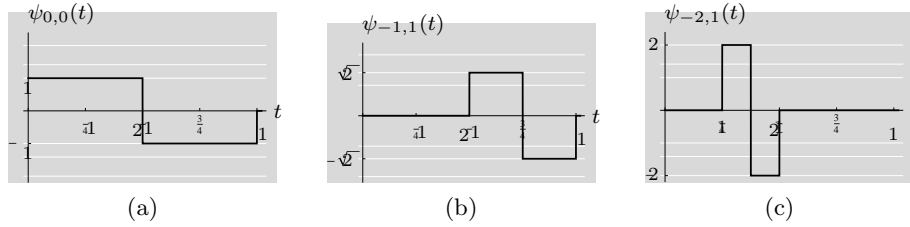
$$\varphi_{\ell,k}(t) = 2^{-\ell/2} \varphi(2^{-\ell}t - k) = \frac{1}{2^{\ell/2}} \varphi\left(\frac{t - 2^\ell k}{2^\ell}\right). \quad (12.16b)$$

A few of the wavelets are given in Figure 12.7. Since we will show that the Haar wavelets form an orthonormal basis, we can define the Haar wavelet series to be

$$\beta_k^{(\ell)} = \langle x, \psi_{\ell,k} \rangle = \int_{-\infty}^{\infty} x(t) \psi_{\ell,k}(t) dt, \quad \ell, k \in \mathbb{Z}, \quad (12.17a)$$

and the inverse Haar wavelet series

$$x(t) = \sum_{\ell \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} \beta_k^{(\ell)} \psi_{\ell,k}(t). \quad (12.17b)$$



**Figure 12.7:** Example Haar basis functions. (a) The prototype function  $\psi(t) = \psi_{0,0}(t)$ ; (b)  $\psi_{-1,1}(t)$ ; (c)  $\psi_{-2,1}(t)$ . (Repeated Figure 0.2.)

We call  $\beta^{(\ell)}$  the *wavelet coefficients*, and denote such a wavelet series pair by

$$x(t) \xleftrightarrow{\text{WS}} \beta_k^{(\ell)}.$$

### Properties of the Haar Wavelet Series

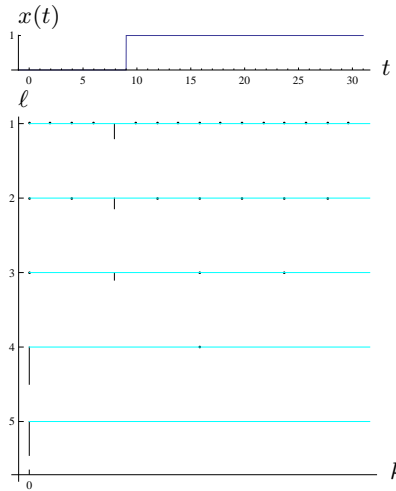
- (i) *Linearity:* The Haar wavelet series operator is a linear operator.
- (ii) *Parseval's Equality:* The Haar wavelet series operator is a unitary operator and thus preserves the Euclidean norm (see (1.51)):

$$\|x\|^2 = \int_{-\infty}^{\infty} |x(t)|^2 dt = \sum_{\ell \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} |\beta_k^{(\ell)}|^2. \quad (12.18)$$

- (iii) *Zero-Moment Property:* We have seen earlier that, while the Haar scaling function with its integer shifts can reproduce constant functions, the Haar wavelet with its integer shifts annihilates them. As the wavelet series uses wavelets as its basis functions, it inherits that property; it annihilates constant functions. In the remainder of the chapter, we will see this to be true for higher-order polynomials with different wavelets.
- (iv) *Characterization of Singularities:* One of the powerful properties of wavelet-like representations is that they can characterize the type and position of singularities via the behavior of wavelet coefficients. Assume, for example, that we want to characterize the step singularity present in the Heaviside function (3.8) with the step at location  $t_0$ . We compute the wavelet coefficient  $\beta_k^{(\ell)}$  to get

$$\begin{aligned} \beta_k^{(\ell)} &= \int_{-\infty}^{\infty} x(t) \psi_{\ell,k}(t) dt \\ &= \begin{cases} 2^{\ell/2}k - 2^{-\ell/2}t_0, & 2^\ell k \leq t_0 < 2^\ell(k + \frac{1}{2}); \\ -2^{\ell/2}(k+1) + 2^{-\ell/2}t_0, & 2^\ell(k + \frac{1}{2}) \leq t_0 < 2^\ell(k+1). \end{cases} \end{aligned}$$

Because the Haar wavelets at a fixed scale do not overlap, there exists exactly one nonzero wavelet coefficient per scale, the one that straddles the discontinuity. Therefore, as  $\ell \rightarrow -\infty$ , the wavelet series zooms towards the singularity,



**Figure 12.8:** Behavior of Haar wavelet coefficients across scales. We plot  $\beta_k^{(\ell)}$  for  $\ell = 1, 2, \dots, 5$ , where  $k$  is dependent on the scale. Because the wavelet is Haar, there is exactly one nonzero coefficient per scale, the one corresponding to the wavelet that straddles the discontinuity.

shown in Figure 12.8. We see that as  $\ell$  decreases from 5 to 1, the single nonzero wavelet coefficients gets closer and closer to the discontinuity.

**Multiresolution Analysis** In the discrete-time Chapters 7-10, we have often encountered coarse/detail approximating spaces  $V$  and  $W$ . We now use the same intuition and start from similar spaces to build the Haar wavelet series in reverse. What we will see is how the iterative construction and the two-scale equations are the manifestations of a fundamental embedding property explicit in the multiresolution analysis of Mallat and Meyer.

We call  $V^{(0)}$  the space of piecewise-constant functions over unit intervals, that is, we say that  $x(t) \in V^{(0)}$ , if and only if  $x(t)$  is constant for  $t \in [k, k+1)$ , and  $x(t)$  is of finite  $\mathcal{L}^2$  norm. Another way to phrase the above is to note that

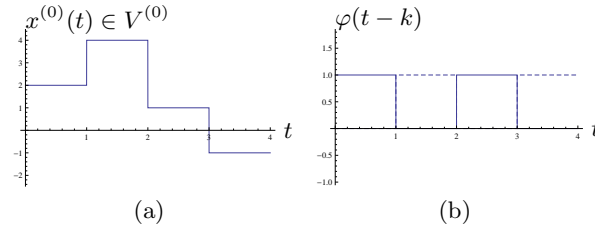
$$V^{(0)} = \text{span}(\{\varphi(t-k)\}_{k \in \mathbb{Z}}) = \text{span}(\{\varphi_{0,k}\}_{k \in \mathbb{Z}}), \quad (12.19)$$

where  $\varphi(t)$  is the Haar scaling function (12.6), and, since  $\langle \varphi(t-k), \varphi(t-m) \rangle = \delta_{k-m}$ , this scaling function and its integer translates form an orthonormal basis for  $V^{(0)}$  (see Figure 12.9). Thus,  $x(t)$  from  $V^{(0)}$  can be written as a linear combination

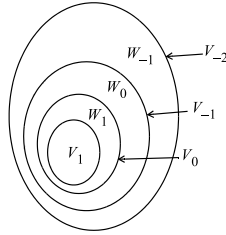
$$x(t) = \sum_{k \in \mathbb{Z}} \alpha_k^{(0)} \varphi(t-k),$$

where  $\alpha_k^{(0)}$  is simply the value of  $x(t)$  on the interval  $[k, k+1)$ . Since  $\|\varphi(t)\| = 1$ ,

$$\|x(t)\|^2 = \|\alpha_k^{(0)}\|^2,$$



**Figure 12.9:** Haar multiresolution spaces and basis functions. (a) A function  $x^{(0)}(t) \in V^{(0)}$ . (b) Basis functions for  $V^{(0)}$ .



**Figure 12.10:** Multiresolution spaces.

or, Parseval's equality for this orthonormal basis. We now introduce a scaled version of  $V^{(0)}$  called  $V^{(\ell)}$ , the space of piecewise-constant functions over intervals of size  $2^\ell$ , that is  $[2^\ell k, 2^\ell(k+1))$ ,  $\ell \in \mathbb{Z}$ . Then,

$$V^{(\ell)} = \text{span}(\{\varphi_{\ell,k}\}_{k \in \mathbb{Z}}),$$

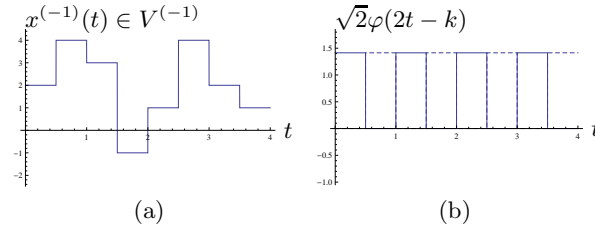
for  $\ell \in \mathbb{Z}$ . For  $\ell > 0$ ,  $V^{(\ell)}$  is a stretched version of  $V^{(0)}$ , and for  $\ell < 0$ ,  $V^{(\ell)}$  is a compressed version of  $V^{(0)}$  (both by  $2^\ell$ ). Moreover, the fact that functions constant over  $[2^\ell k, 2^\ell(k+1))$  are also constant over  $[2^m k, 2^m(k+1))$ ,  $\ell > m$ , leads to the *inclusion property* (see Figure 12.10),

$$V^{(\ell)} \subset V^{(m)} \quad \ell > m. \quad (12.20)$$

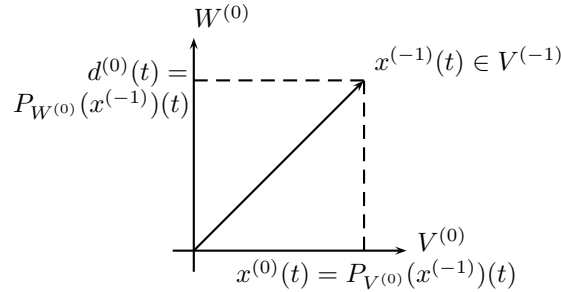
We can use this to derive the two-scale equation (12.8), by noting that because of  $V^{(0)} \subset V^{(-1)}$ ,  $\varphi(t)$  can be expanded in the basis for  $V^{(-1)}$ . Graphically, we show the spaces  $V^{(0)}$ ,  $V^{(-1)}$ , and their basis functions in Figures 12.9 and Figure 12.11; the two-scale equation was shown in Figure 12.3.

What about the detail spaces? Take a function  $x^{(-1)}(t)$  in  $V^{(-1)}$  but not in  $V^{(0)}$ ; such a function is constant over half-integer intervals but not so over integer intervals (see Figure 12.11(a)). Decompose it as a sum of its projections onto  $V^{(0)}$  and  $W^{(0)}$ , the latter the orthogonal complement of  $V^{(0)}$  in  $V^{(-1)}$  (see Figure 12.12),

$$x^{(-1)}(t) = P_{V^{(0)}}(x^{(-1)})(t) + P_{W^{(0)}}(x^{(-1)})(t). \quad (12.21)$$



**Figure 12.11:** Haar multiresolution spaces and basis functions. (a) A function  $x^{(-1)}(t) \in V^{(-1)}$ . (b) Basis functions for  $V^{(-1)}$ .



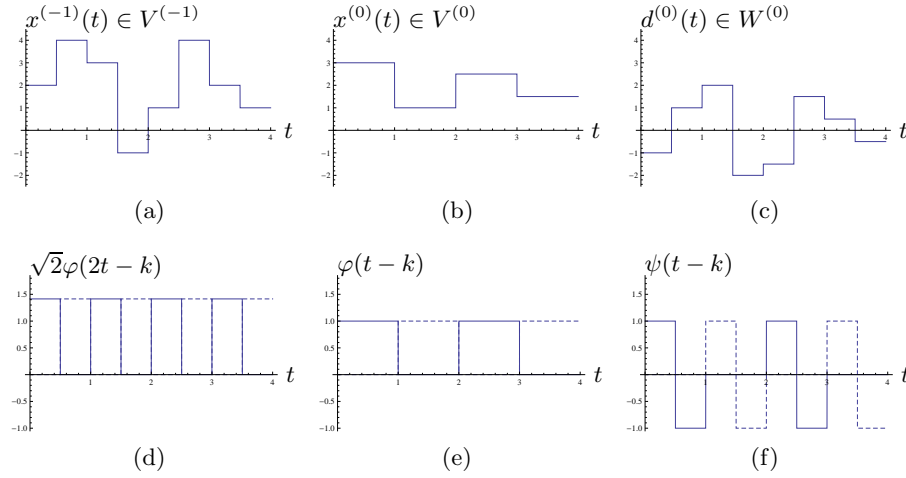
**Figure 12.12:** A function from  $V^{(1)}$  as the sum of its projections onto  $V^{(0)}$  and  $W^{(0)}$ .

We first find the projection of  $x^{(-1)}(t)$  onto  $V^{(0)}$  as

$$\begin{aligned}
 P_{V^{(0)}}(x^{(-1)})(t) &= x^{(0)}(t), \\
 &= \sum_{k \in \mathbb{Z}} \alpha_k^{(0)} \varphi(t - k) = \sum_{k \in \mathbb{Z}} \langle x^{(-1)}(t), \varphi(t - k) \rangle_t \varphi(t - k), \\
 &\stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} \langle x^{(-1)}(t), \varphi(2t - 2k) + \varphi(2t - 2k - 1) \rangle_t \varphi(t - k), \\
 &\stackrel{(b)}{=} \sum_{k \in \mathbb{Z}} \frac{1}{2} \left[ x^{(-1)}(k) + x^{(-1)}(k + \frac{1}{2}) \right] \varphi(t - k), \tag{12.22}
 \end{aligned}$$

where (a) follows from the two-scale equation (12.8); and (b) from evaluating the inner product between  $x^{(-1)}(t)$  and the basis functions for  $V^{(-1)}$ . In other words,  $x^{(0)}(t)$  is simply the average of  $x^{(-1)}(t)$  over two successive intervals. This is the best least-squares approximation of  $x^{(-1)}(t)$  by a function in  $V^{(0)}$  (see Exercise 12.5).

We now find the projection of  $x^{(-1)}(t)$  onto  $W^{(0)}$ . Subtract the projection  $x^{(0)}$  from  $x^{(-1)}$  and call the difference  $d^{(0)}$ . Since  $x^{(0)}$  is an orthogonal projection,  $d^{(0)}$



**Figure 12.13:** Haar decomposition of a function (a)  $x^{(-1)}(t) \in V^{(-1)}$  into a projection (b)  $x^{(0)}(t) \in V^{(0)}$  (average over two successive intervals) and (c)  $d^{(0)}(t) \in W^{(0)}$  (difference over two successive intervals). (d)–(f) Appropriate basis functions.

is orthogonal to  $V^{(0)}$  (see Figure 12.12). Using (12.22) leads to

$$\begin{aligned}
 P_{W^{(0)}}(x^{(-1)})(t) &= d^{(0)}(t) = x^{(-1)}(t) - x^{(0)}(t), \\
 &= \begin{cases} \frac{1}{2} [x^{(-1)}(k) - x^{(-1)}(k + \frac{1}{2})], & k \leq t < k + \frac{1}{2}; \\ -\frac{1}{2} [x^{(-1)}(k) - x^{(-1)}(k + \frac{1}{2})], & k + \frac{1}{2} \leq t < k + 1, \end{cases} \\
 &= \sum_k \frac{1}{2} [x^{(-1)}(k) - x^{(-1)}(k + \frac{1}{2})] \psi(t - k) \\
 &= \sum_k \frac{1}{\sqrt{2}} [\beta_{2k}^{(-1)} - \beta_{2k+1}^{(-1)}] \psi(t - k) = \sum_k \beta_k^{(0)} \psi(t - k). \quad (12.23)
 \end{aligned}$$

We have thus informally shown that the space  $V^{(-1)}$  can be decomposed as

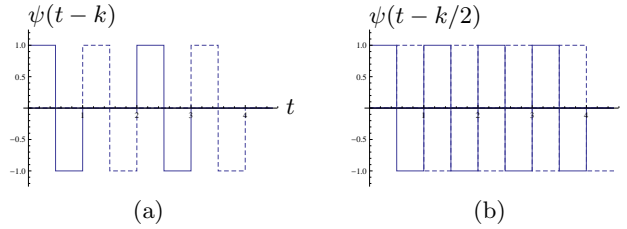
$$V^{(-1)} = V^{(0)} \oplus W^{(0)} \quad (12.24)$$

(see an example in Figure 12.13). We also derived bases for these spaces, scaling functions and their shifts for  $V^{(-1)}$  and  $V^{(0)}$ , and wavelets and their shifts for  $W^{(0)}$ . This process can be further iterated on  $V_0$  (see Figure 12.10).

### 12.1.3 Haar Frame Series

The Haar wavelet series we just saw is an elegant representation, and completely nonredundant. As we have seen in Chapters 10 and 11, at times we can benefit from relaxing this constraint, and allowing some redundancy in the system. Our aim would then be to build frames. There are many ways in which we could do that. For example, by adding to the Haar wavelet basis wavelets at points halfway





**Figure 12.14:** A few Haar wavelets at scale  $\ell = 0$  for the (a) Haar wavelet series (nonredundant) and (b) Haar frame series (redundant). Clearly, there are twice as many wavelets in (b), making it a redundant expansion with the redundancy factor 2.

in between the existing ones, we would have twice as many wavelets, leading to a redundant series representation with a redundancy factor of 2, a simple example we will use to illustrate Haar frame series.

**Definition of the Haar Frame Series** We now relax the constraint of critical sampling, but still retain the series expansion. That is, we assume that expansion coefficients are

$$\beta_k^{(\ell)} = \langle x, \psi_{\ell,k} \rangle = \int_{-\infty}^{\infty} x(t) \psi_{\ell,k}(t) dt, \quad \ell, k \in \mathbb{Z}, \quad (12.25)$$

where  $\psi_{\ell,k}(t)$  is now given by

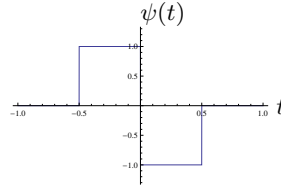
$$\psi_{\ell,k}(t) = a_0^{-\ell/2} \psi(a_0^{-\ell} t - b_0 k) = a_0^{-\ell/2} \psi\left(\frac{t - a_0^{\ell} b_0 k}{a_0^{\ell}}\right), \quad (12.26)$$

with  $a_0 > 1$  and  $b_0 > 0$ . With  $a_0 = 2$  and  $b_0 = 1$ , we get back our nonredundant Haar wavelet series. What we have allowed ourselves to do here is to choose different scale factors from 2, as well as different coverage of the time axis by shifted wavelets at a fixed scale. For example, keep the scale factor the same,  $a_0 = 2$ , but allow overlap of half width between Haar wavelets, that is, choose  $b_0 = 1/2$ . Figure 12.14(b) shows how many wavelets then populate the time axis at a fixed scale (example for  $\ell = 0$ ), compared to the wavelet series (part (a) of the same figure).

**Properties of the Haar Frame Series** Such a Haar frame series satisfies similar properties as the Haar wavelet series: it is linear, it is able to characterize singularities, it inherits the zero-moment property. One property though, Parseval's equality, bears further scrutiny. Let us express the energy of the expansion coefficients as:

$$\sum_{\ell \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} |\beta_{k/2}^{(\ell)}|^2 = \sum_{\ell \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} |\beta_k^{(\ell)}|^2 + \sum_{\ell \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} |\beta_{k+1/2}^{(\ell)}|^2 \stackrel{(a)}{=} 2 \|x\|^2,$$

where (a) follows from (12.18). In other words, this frame series behaves like two orthonormal bases glued together; it is then not surprising that the energy in the expansion coefficients is twice that of the input function, making this transform a tight frame.



**Figure 12.15:** The prototype wavelet in the Haar continuous wavelet transform.

### 12.1.4 Haar Continuous Wavelet Transform

We finally relax all the constraints and discuss the most redundant version of a wavelet expansion, where the Haar wavelet (12.12) is now shifted to be centered at  $t = 0$  (see Figure 12.15):

$$\psi(t) = \begin{cases} 1, & -1/2 \leq t < 0; \\ -1, & 0 \leq t < 1/2; \\ 0, & \text{otherwise.} \end{cases} \quad (12.27)$$

Then, instead of  $a = a_0^\ell$ , we allow all positive real numbers,  $a \in \mathbb{R}^+$ . Similarly, instead of shifts  $b = b_0 k$ , we allow all real numbers,  $b \in \mathbb{R}$ :

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right), \quad a \in \mathbb{R}^+, \quad b \in \mathbb{R}, \quad (12.28)$$

with  $\psi(t)$  the Haar wavelet. The scaled and shifted Haar wavelet is then centered at  $t = b$  and scaled by a factor  $a$ . All the wavelets are again of unit norm,  $\|\psi_{a,b}(t)\| = 1$ . For  $a = 2^\ell$  and  $b = 2^\ell k$ , we get the nonredundant wavelet basis as in TBD.

**Definition of the Haar Continuous Wavelet Transform** We then define the Haar continuous wavelet transform to be (an example is given in Figure 12.16):

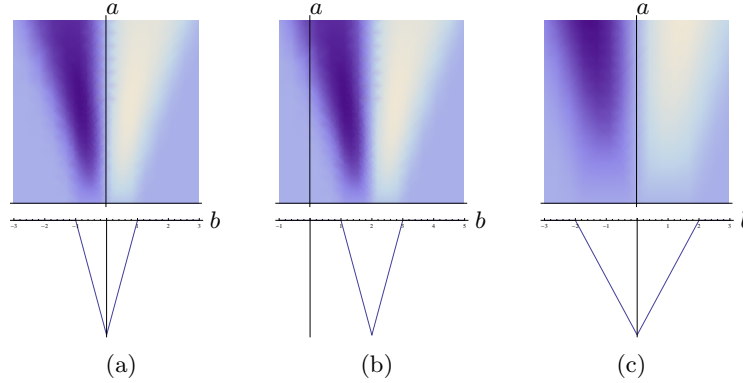
$$X(a, b) = \langle x, \psi_{a,b} \rangle = \int_{-\infty}^{\infty} x(t) \psi_{a,b}(t) dt, \quad a \in \mathbb{R}^+, \quad b \in \mathbb{R}, \quad (12.29a)$$

with  $\psi_{a,b}(t)$  from (12.28), with the inverse Haar continuous wavelet transform

$$x(t) = \frac{1}{C_\psi} \int_{a \in \mathbb{R}^+} \int_{b \in \mathbb{R}} X(a, b) \psi_{a,b}(t) \frac{db da}{a^2}, \quad (12.29b)$$

where the equality holds in  $\mathcal{L}^2$  sense. To denote such a pair, we write:

$$x(t) \xleftrightarrow{\text{CWT}} X(a, b).$$



**Figure 12.16:** (a) The Haar wavelet transform of an example function  $x(t)$  (hat function). (b) Illustration of the shift-in-time property for  $x(t - 2)$ . (c) Illustration of the scaling-in-time property for  $x(t/2)$ .

### Properties of the Haar Continuous Wavelet Transform

- (i) *Linearity:* The Haar continuous wavelet transform operator is a linear operator.
- (ii) *Shift in time:* A shift in time by  $t_0$  results in (see Figure 12.16(b))

$$x(t - t_0) \xleftrightarrow{\text{CWT}} X(a, b - t_0). \quad (12.30)$$

- (iii) *Scaling in time:* Scaling in time by  $\alpha$  results in (see Figure 12.16(c))

$$x(\alpha t) \xleftrightarrow{\text{CWT}} \frac{1}{\sqrt{\alpha}} X\left(\frac{a}{\alpha}, \frac{b}{\alpha}\right). \quad (12.31)$$

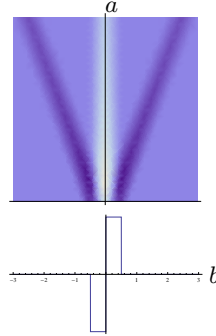
- (iv) *Parseval's equality:* Parseval's equality holds for the Haar continuous wavelet transform; we omit it here and revisit it in the general context in (12.112).
- (v) *Redundancy:* Just like for the local Fourier transform, the continuous wavelet transform maps a function of one variable into a function of two variables. It is thus highly redundant, and this redundancy is expressed by the *reproducing kernel*:

$$K(a_0, b_0, a, b) = \langle \psi_{a_0, b_0}, \psi_{a, b} \rangle, \quad (12.32)$$

a four-dimensional function. Figure 12.17 shows the reproducing kernel of the Haar wavelet, namely  $K(1, 0, a, b)$ ; note that the reproducing kernel is zero at all dyadic scale points as the wavelets are then orthogonal to each other.

- (vi) *Characterization of Singularities:* This is one of the most important properties of the continuous wavelet transform, since, by looking at its behavior, we can infer the type and position of singularities occurring in the function.

For example, assume we are given a Dirac delta function at location  $t_0$ ,  $x(t) = \delta(t - t_0)$ . At scale  $a$ , only those Haar wavelets whose support straddles



**Figure 12.17:** The Haar wavelet transform of the Haar wavelet (12.12). This is also the reproducing kernel,  $K(1, 0, a, b)$ , of the Haar wavelet.

$t_0$  will produce nonzero coefficients. Because of the form of the Haar wavelet, the nonzero coefficients will extend over a region of size  $2^a$  around  $t_0$ . As  $a \rightarrow -\infty$ , these coefficients focus arbitrarily closely on the singularity. Moreover, these coefficients grow at a specific rate, another way to identify the type of a singularity. We will go into more details on this later in the chapter.

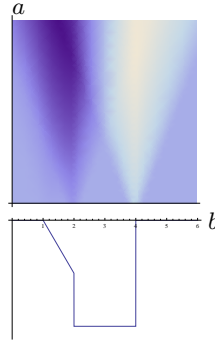
As a simple example, take  $x(t)$  to be the Heaviside function (3.8) with the step at location  $t_0$ . We want to see how the Haar wavelet (12.27) isolates and characterizes the step singularity. To do that, we will need two things: (1) The primitive of the wavelet, defined as

$$\theta(t) = \int_{-\infty}^t \psi(\tau) d\tau = \begin{cases} 1/2 - |t|, & |t| < 1/2; \\ 0, & \text{otherwise,} \end{cases} \quad (12.33)$$

that is, a hat function (3.49a) on the interval  $|t| < 1/2$ . Note that the primitive of the scaled and normalized wavelet  $a^{-1/2}\psi(t/a)$  is  $\sqrt{a}\theta(t/a)$ , or a factor  $a$  larger due to integration. (2) We also need the derivative of  $x(t)$ , which exists only in a generalized sense (using distributions) and can be shown to be a Dirac delta function at  $t_0$ ,  $x'(t) = \delta(t - t_0)$ .

Now, the continuous wavelet transform of the Heaviside function follows as

$$\begin{aligned} X(a, b) &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) x(t) dt \\ &\stackrel{(a)}{=} \left[ \sqrt{a} \theta\left(\frac{t-b}{a}\right) x(t) \right]_{-\infty}^{\infty} - \int_{-\infty}^{\infty} \sqrt{a} \theta\left(\frac{t-b}{a}\right) x'(t) dt \\ &\stackrel{(b)}{=} - \int_{-\infty}^{\infty} \sqrt{a} \theta\left(\frac{t-b}{a}\right) \delta(t - t_0) dt \\ &\stackrel{(c)}{=} \sqrt{a} \theta\left(\frac{t_0 - b}{a}\right), \end{aligned} \quad (12.34)$$



**Figure 12.18:** The Haar wavelet transform of a piecewise-polynomial function  $x(t)$  as in (12.35).

where (a) follows from integration by parts; (b) from  $\theta$  being of compact support; and (c) from the shifting property of the Dirac delta function in Table 3.1. Thus, as  $a \rightarrow 0$ , the continuous wavelet transform zooms towards the singularity and scales as  $a^{1/2}$ , with a shape given by the primitive of the wavelet  $\theta(t)$ ; thus, we may expect a hat-like region of influence (see Figure 12.18 at the step discontinuity,  $t = 4$ , for illustration).

This discussion focused on the behavior of the Haar wavelet transform around a point of singularity; what about smooth regions? Take  $x(t)$  to be

$$x(t) = \begin{cases} t-1, & 1 \leq t < 2; \\ 2, & 2 \leq t < 4; \\ 0, & \text{otherwise,} \end{cases} \quad (12.35)$$

and the Haar wavelet (12.27). The function  $x(t)$  has three singularities, discontinuities at  $t = 1, 2, 3$ . The wavelet has 1 zero moment, so it will have a zero inner product inside the interval  $[2, 3]$ , where  $x(t)$  is constant (see Figure 12.18). What happens in the interval  $[1, 2]$ , where  $x(t)$  is linear?

Calculate the continuous wavelet transform for some shift  $b \in [1, 2]$  for  $a$  sufficiently small so that the support of the shifted wavelet  $[b - a/2, b + a/2] \in [1, 2]$ :

$$X(a, b) = \frac{1}{\sqrt{a}} \left( \int_{b-a/2}^b t dt - \int_b^{b+a/2} t dt \right) = -\frac{1}{4} a^{3/2}. \quad (12.36)$$

Thus, the lack of a second zero moment (which would have produced a zero inner product) produces a residual of order  $a^{3/2}$  as  $a \rightarrow 0$ .

To study qualitatively the overall behavior, there will be two cones of influence at singular points 2 and 3, with an order  $a^{1/2}$  behavior as in (12.34), a constant of order  $a^{3/2}$  in the  $(1, 2)$  interval as in (12.36) (which spills over into the  $(2, 3)$  interval), and zero elsewhere, shown in Figure 12.18.

### Chapter Outline

We now follow the path set through this simple Haar example, and follow with more general developments. We start in Section 12.2 with iterated filter banks building scaling functions and wavelets as limits of iterated filters. We study issues of convergence and smoothness of resulting functions. In Section 12.3, we then define and look into the properties of wavelet series: localization, zero moments and decay of wavelet coefficients, before considering the characterization of singularities by the decay of the associated wavelet series coefficients. We study multiresolution analysis, revisiting the wavelet construction from an axiomatic point of view. In Section 12.4, we relax the constraints of nonredundancy to construct wavelet frames, midway between the nonredundant wavelet series and a completely redundant wavelet transform. We follow in Section 12.5 with the continuous wavelet transform. Section 12.6 is devoted to computational issues, in particular to Mallat's algorithm, which allows us to compute wavelet coefficients with an initial continuous-time projection followed by a discrete-time, filter-bank algorithm.

*Notation used in this chapter:* In most of this chapter, we consider real-valued wavelets only and thus the domain for the scale factor  $a$  is  $\mathbb{R}^+$ ; the extension to complex wavelets requires simply  $a \in \mathbb{R}$ ,  $a \neq 0$ .  $\square$

## 12.2 Scaling Function and Wavelets from Orthogonal Filter Banks

In the previous section, we set the stage for this section by examining basic properties of iterated filter banks with Haar filters. The results in this section should thus not come as a surprise, as they generalize those for Haar filter banks.

We start with an orthogonal filter bank with filters  $g$  and  $h$  whose properties were summarized in Table 7.9, Chapter 7, where we used these filters and their shifts by multiples by two as the basis for  $\ell^2(\mathbb{Z})$ . This orthonormal basis was implemented using a critically-sampled two-channel filter bank with down- and upsampling by 2, an orthogonal synthesis lowpass/highpass filter pair  $g_n, h_n$  and a corresponding analysis lowpass/highpass filter pair  $g_{-n}, h_{-n}$ . We then used these filters and the associated two-channel filter bank as building blocks for a DWT in Chapter 9. For example, we saw that in a 3-level iterated Haar filter bank, the lowpass and highpass at level 3 were given by (9.1c)–(9.1d) and plotted in Figure 9.4; we repeated the last-level filters in (12.2). Another example, a 3-level iterated filter bank with Daubechies filters, was given in Example 9.1.

### 12.2.1 Iterated Filters

As for the Haar case, we come back to filters and their iterations, and associate a continuous-time function to the discrete-time sequence representing the impulse response of the iterated filter.

We assume a length- $L$  orthogonal lowpass/highpass filter pair  $(g_n, h_n)$ , and

## 12.2. Scaling Function and Wavelets from Orthogonal Filter Banks

807

write the equivalent filters at the last level of a  $J$ -level iterated filter bank:

$$G^{(J)}(z) = \prod_{\ell=0}^{J-1} G(z^{2^\ell}) = G^{(J-1)}(z) G(z^{2^{J-1}}), \quad (12.37a)$$

$$H^{(J)}(z) = \prod_{\ell=0}^{J-2} G(z^{2^\ell}) H(z^{2^{J-1}}) = G^{(J-1)}(z) H(z^{2^{J-1}}). \quad (12.37b)$$

We know that, by construction, these filters are orthonormal to their shifts by  $2^J$ , (9.6a), (9.11a), as well as orthogonal to each other, (9.14a).

The equivalent filters  $g^{(J)}$ ,  $h^{(J)}$  have norm 1 and length  $L^{(J)}$ , which can be upper bounded by (see (9.5b))

$$L^{(J)} \leq (L-1)2^J. \quad (12.38)$$

## 12.2.2 Scaling Function and its Properties

We now associate a piecewise-constant function  $\varphi^{(J)}(t)$  to  $g_n^{(J)}$  so that  $\varphi^{(J)}(t)$  is of finite length and norm 1. Since the number of piecewise segments (equal to the number of nonzero coefficients of  $g_n^{(J)}$ ) grows exponentially with  $J$  (see (12.38)), we choose their width as  $2^{-J}$ , upper bounding the length of  $\varphi^{(J)}(t)$  by  $(L-1)$ :

$$\text{support}(\varphi^{(J)}(t)) \subset [0, L-1], \quad (12.39)$$

where  $\text{support}(\cdot)$  stands for the interval of the real line where the function is different from zero. For  $\varphi^{(J)}(t)$  to inherit the unit-norm property from  $g_n^{(J)}$ , we choose the height of the piecewise segments as  $2^{J/2}g_n^{(J)}$ . Then, the  $n$ th piece of the  $\varphi^{(J)}(t)$  contributes

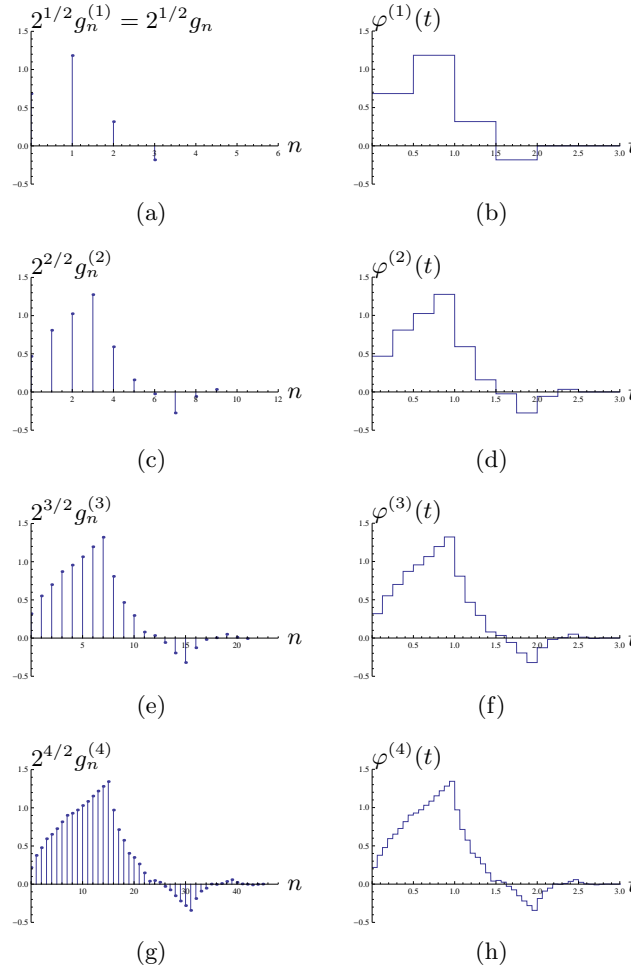
$$\int_{n/2^J}^{(n+1)/2^J} |\varphi^{(J)}(t)|^2 dt = \int_{n/2^J}^{(n+1)/2^J} 2^J (g_n^{(J)})^2 dt = (g_n^{(J)})^2$$

to  $\varphi^{(J)}(t)$ . Summing up the individual contributions,

$$|\varphi^{(J)}(t)|^2 = \sum_{n=0}^{L^{(J)}-1} \int_{n/2^J}^{(n+1)/2^J} |\varphi^{(J)}(t)|^2 dt = \sum_{n=0}^{L^{(J)}-1} (g_n^{(J)})^2 = 1.$$

We have thus defined the piecewise-constant function as

$$\begin{aligned} \varphi^{(J)}(t) &= 2^{J/2} g_n^{(J)}, & \frac{n}{2^J} &\leq t < \frac{n+1}{2^J}, \\ &= \sum_{n=0}^{L^{(J)}-1} g_n^{(J)} 2^{J/2} \varphi_h(2^J t - n), \end{aligned} \quad (12.40)$$



**Figure 12.19:** Iterated filter  $2^{J/2} g_n^{(J)}$  and associated piecewise-constant function  $\varphi^{(J)}(t)$  based on a 4-tap Daubechies lowpass filter (9.10) at level (a)–(b)  $J = 1$ ; (c)–(d)  $J = 2$ ; (e)–(f)  $J = 3$ ; and (g)–(h)  $J = 4$ . Note that we have rescaled the equivalent filters' impulse responses as well as plotted them at different discrete intervals to highlight the correspondences with their piecewise-constant functions.

where  $\varphi_h(t)$  is the Haar scaling function (box function) from (12.6). We verified that the above iterated function is supported on a finite interval and has unit norm.<sup>154</sup>

In Figure 12.19, we show a few iterations of a 4-tap filter from Example 9.1 and its associated piecewise-constant function. The piecewise-constant function  $\varphi^{(J)}$  has geometrically decreasing piecewise segments and a support contained in

<sup>154</sup>We could have defined a piecewise-linear function instead of a piecewise-constant one, but it does not change the behavior of the limit we will study.



## 12.2. Scaling Function and Wavelets from Orthogonal Filter Banks

809

the interval  $[0, 3]$ . From the figure it is clear that the *smoothness* of  $\varphi^{(J)}(t)$  depends on the *smoothness* of  $g_n^{(J)}$ . If the latter tends, as  $J$  increases, to a sequence with little local variation, then the piecewise-constant approximation will tend to a smooth function as well, as the piecewise segments become finer and finer. On the contrary, if  $g_n^{(J)}$  has too much variation as  $J \rightarrow \infty$ , the sequence of functions  $\varphi^{(J)}(t)$  might not have a limit as  $J \rightarrow \infty$ . This leads to the following necessary condition for the filter  $g_n$ , the proof of which is given in Solved Exercise 12.1:

**PROPOSITION 12.1 (NECESSITY OF A ZERO AT  $\pi$ )** For the  $\lim_{J \rightarrow \infty} \varphi^{(J)}(t)$  to exist, it is necessary for  $G(e^{j\omega})$  to have a zero at  $\omega = \pi$ .

As a direct corollary of this result, the necessity of a zero at  $\omega = \pi$  translates also to the necessity of

$$G(e^{j\omega})|_{\omega=\pi} = \sqrt{2}, \quad (12.41)$$

because of (7.13). We are now ready to define the limit function:

**DEFINITION 12.2 (SCALING FUNCTION)** We call the scaling function  $\varphi(t)$  the limit, when it exists, of:

$$\varphi(t) = \lim_{J \rightarrow \infty} \varphi^{(J)}(t), \quad (12.42)$$

**Scaling Function in the Fourier Domain** We now find the Fourier transform of  $\varphi^{(J)}(t)$ , denoted by  $\Phi^{(J)}(\omega)$ . The functions  $\varphi^{(J)}(t)$  is a linear combination of box functions, each of width  $1/2^J$  and height  $2^{J/2}$ , where the unit box function is equal to the Haar scaling function (12.6), with the Fourier transform  $\Phi_h(\omega)$  as in (12.7). Using the scaling-in-time property of the Fourier transform (3.58a), the transform of a box function on the interval  $[0, 1/2^J]$  of height  $2^{J/2}$  is

$$\Phi_h^{(J)}(\omega) = 2^{-J/2} e^{-j\omega/2^{J+1}} \frac{\sin(\omega/2^{J+1})}{\omega/2^{J+1}}. \quad (12.43)$$

Shifting the  $n$ th box to start at  $t = n/2^J$  multiplies its Fourier transform by  $e^{-j\omega n/2^J}$ . Putting it all together, we find

$$\begin{aligned} \Phi^{(J)}(\omega) &= \Phi_h^{(J)}(\omega) \sum_{n=0}^{L^{(J)}-1} e^{-j\omega n/2^J} g_n^{(J)} \stackrel{(a)}{=} \Phi_h^{(J)}(\omega) G^{(J)}(e^{j\omega/2^J}) \\ &\stackrel{(b)}{=} \Phi_h^{(J)}(\omega) \prod_{\ell=0}^{J-1} G(e^{j\omega 2^\ell/2^J}) \stackrel{(c)}{=} \Phi_h^{(J)}(\omega) \prod_{\ell=1}^J G(e^{j\omega/2^\ell}), \end{aligned} \quad (12.44)$$

where (a) follows from the definition of the DTFT (2.78a); (b) from (12.3); and (c) from reversing the order of the factors in the product.

In the sequel, we will be interested in what happens in the limit, when  $J \rightarrow \infty$ . For any finite  $\omega$ , the effect of the interpolation function  $\Phi_h^{(J)}(\omega)$  becomes negligible as  $J \rightarrow \infty$ . Indeed in (12.43), both terms dependent on  $\omega$  tend to 1 as  $J \rightarrow \infty$  and only the factor  $2^{-J/2}$  remains. So, in (12.44), the key term is the product, which becomes an infinite product, which we now define:

**DEFINITION 12.3 (FOURIER TRANSFORM OF THE INFINITE PRODUCT)** We call  $\Phi(\omega)$  the limit, if it exists, of the infinite product:

$$\Phi(\omega) = \lim_{J \rightarrow \infty} \Phi^{(J)}(\omega) = \prod_{\ell=1}^{\infty} 2^{-1/2} G(e^{j\omega/2^\ell}). \quad (12.45)$$

The corollary to Proposition 12.1, (12.41) is now clear; if  $G(1) > \sqrt{2}$ ,  $\Phi(0)$  would grow unbounded, and if  $G(1) < \sqrt{2}$ ,  $\Phi(0)$  would be zero, contradicting the fact that  $\Phi(\omega)$  is the limit of lowpass filters and hence a lowpass function.

A more difficult question is to understand when the limits of the time-domain iteration  $\varphi^{(J)}(t)$  (12.40) and the Fourier-domain iteration  $\Phi^{(J)}(\omega)$  (12.44) form the Fourier-transform pair. We will show in Example 12.1 that this is a nontrivial question. As the exact conditions are technical and beyond the scope of our text, we concentrate on those cases when the limits in Definitions 12.2 and 12.3 are well defined and form a Fourier-transform pair, that is, when

$$\varphi(t) \xleftrightarrow{\text{FT}} \Phi(\omega),$$

We now look into the behavior of the infinite product. If  $\Phi(\omega)$  decays sufficiently fast in  $\omega$ , the scaling function  $\varphi(t)$  will be smooth. How this can be done while maintaining other desirable properties (such as compact support and orthogonality) is the key result for designing wavelet bases from iterated filter banks.

**EXAMPLE 12.1 (FOURIER TRANSFORM OF THE INFINITE PRODUCT)** To gain intuition, we now look into examples of filters and their associated infinite products.

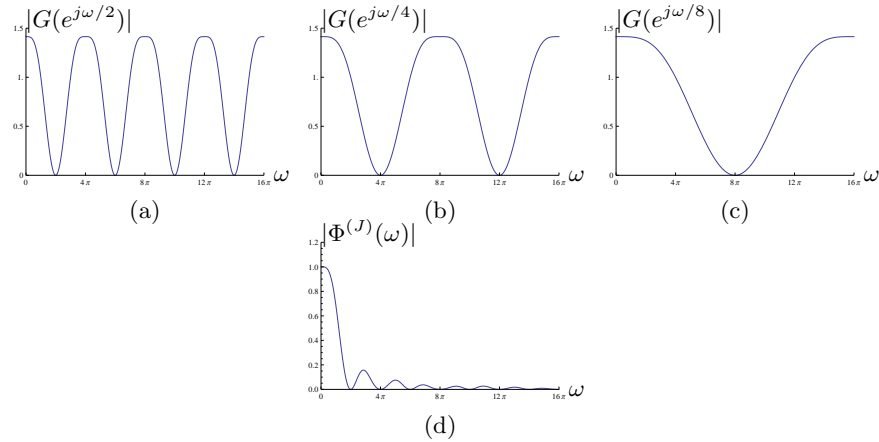
- (i) *Daubechies filter with two zeros at  $\omega = \pi$ :* We continue with our example of the Daubechies lowpass filter from (9.10) with its associated piecewise-constant function in Figure 12.19. In the Fourier-domain product (12.44), the terms are periodic with periods  $4\pi, 8\pi, \dots, 2^J 2\pi$ , since  $G(e^{j\omega})$  is  $2\pi$ -periodic (see Figure 12.20(a)–(c)). We show the product in part (d) of the figure. The terms are oscillating depending on their periodicity, but the product decays rather nicely. We will study this decay in detail shortly.
- (ii) *Length-4 filter designed using lowpass approximation method:* Consider the orthogonal filter designed using the window method in Example 7.2. This filter does not have a zero at  $\omega = \pi$ , since

$$G(e^{j\omega})|_{\omega=\pi} \approx 0.389.$$

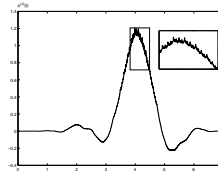
Its iteration is shown in Figure 12.21, with noticeable high-frequency oscillations, prohibiting convergence of the iterated function  $\varphi^{(J)}(t)$ .

## 12.2. Scaling Function and Wavelets from Orthogonal Filter Banks

811



**Figure 12.20:** Factors (a)  $|G(e^{j\omega/2})|$ , (b)  $|G(e^{j\omega/4})|$ , and (c)  $|G(e^{j\omega/8})|$  that appear in (d) the Fourier-domain product  $\Phi^{(J)}(\omega)$ .



**Figure 12.21:** Iteration of a filter without a zero at  $\omega = \pi$ . The high-frequency oscillations prohibit the convergence of the iterated function  $\varphi^{(J)}(t)$ .

(iii) *Stretched Haar filter:* Instead of the standard Haar filter, consider:

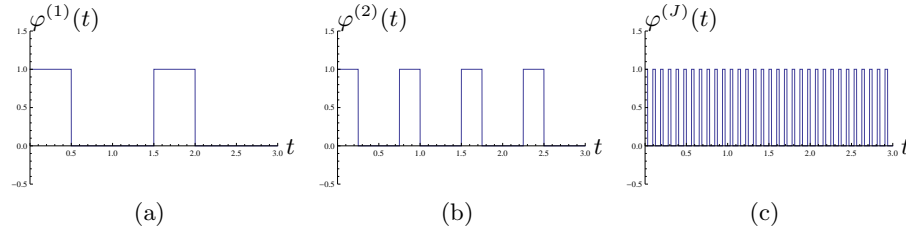
$$g = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \xleftrightarrow{\text{ZT}} G(z) = \frac{1}{\sqrt{2}}(1 + z^{-3}).$$

It is clearly an orthogonal lowpass filter and has one zero at  $\omega = \pi$ . However, unlike the Haar filter, its iteration is highly unsmooth. Consider the equivalent filter after  $J$  stages of iteration:

$$g^{(J)} = \frac{1}{2^{J/2}} \begin{bmatrix} 1 & 0 & 0 & 1 & \dots & 1 & 0 & 0 & 1 \end{bmatrix}.$$

The piecewise-constant function  $\varphi^{(J)}(t)$  inherits this lack of smoothness, and does not converge pointwise to a proper limit, as shown graphically in Figure 12.22. Considering the frequency domain and the infinite product, it turns out that  $\mathcal{L}^2$  convergence fails as well (see Exercise 12.2).

The examples above show that iterated filters and their associated graphical functions behave quite differently. The Haar case we saw in the previous section



**Figure 12.22:** Iteration of the stretched Haar filter with impulse response  $g = [1/\sqrt{2} \ 0 \ 0 \ 1/\sqrt{2}]$ . (a)  $\varphi^{(1)}(t)$ . (b)  $\varphi^{(2)}(t)$ . (c)  $\varphi^{(J)}(t)$ .

was trivial, the 4-tap filters showed a smooth behavior, and the stretched Haar filter pointed out potential convergence problems.

In the sequel, we concentrate on orthonormal filters with  $N \geq 1$  zeros at  $\omega = \pi$ , or

$$G(e^{j\omega}) = \left( \frac{1 + e^{-j\omega}}{2} \right)^N R(e^{j\omega}), \quad (12.46)$$

with  $R(e^{j\omega})|_{\omega=\pi} = \sqrt{2}$  for the limit to exist. We assume (1) pointwise convergence of the iterated function  $\varphi^{(J)}(t)$  to  $\varphi(t)$ , (2) pointwise convergence of the iterated Fourier-domain function  $\Phi^{(J)}(\omega)$  to  $\Phi(\omega)$ , and finally (3) that  $\varphi(t)$  and  $\Phi(\omega)$  are a Fourier-transform pair. In other words, we avoid all convergence issues and concentrate on the well-behaved cases exclusively.

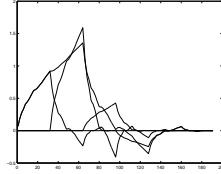
**Two-Scale Equation** We have seen in Section 12.1.1 that the Haar scaling function satisfies a two-scale equation 12.8. This is true in general, except that more terms will be involved in the summation. To show this, we start with the Fourier-domain limit of the infinite product (12.45):

$$\begin{aligned} \Phi(\omega) &= \prod_{\ell=1}^{\infty} 2^{-1/2} G(e^{j\omega/2^\ell}) \stackrel{(a)}{=} 2^{-1/2} G(e^{j\omega/2}) \prod_{\ell=2}^{\infty} 2^{-1/2} G(e^{j\omega/2^\ell}), \\ &\stackrel{(b)}{=} 2^{-1/2} G(e^{j\omega/2}) \Phi(\omega/2) \stackrel{(c)}{=} 2^{-1/2} \sum_{n=0}^{L-1} g_n e^{-j\omega n/2} \Phi(\omega/2), \\ &\stackrel{(d)}{=} \sum_{n=0}^{L-1} g_n \left[ 2^{-1/2} e^{-j\omega n/2} \Phi(\omega/2) \right], \end{aligned} \quad (12.47)$$

where in (a) we took one factor out,  $2^{-1/2} G(e^{j\omega/2})$ ; in (b) we recognize the infinite product as  $\Phi(\omega/2)$ ; (c) follows from the definition of the DTFT (2.78a); and in (d) we just rearranged the terms. Then, using the scaling-in-time property of the Fourier transform (3.58a),  $\Phi(\omega/2) \xrightarrow{\text{FT}} 2\varphi(2t)$ , and the shift-in-time property (3.56),

## 12.2. Scaling Function and Wavelets from Orthogonal Filter Banks

813



**Figure 12.23:** Two-scale equation for the Daubechies scaling function. (a) The scaling function  $\varphi(t)$  and (b) expressed as a linear combination of  $\varphi(2t - n)$ .

$e^{-j\omega n/2} X(\omega) \xrightarrow{\text{FT}} x(t - n/2)$ , we get the two-scale equation:

$$\varphi(t) = \sqrt{2} \sum_{n=0}^{L-1} g_n \varphi(2t - n), \quad (12.48)$$

shown in Figure 12.23 for the Daubechies 4-tap filter from Examples 12.1 and 12.2.

**Smoothness** As seen earlier, the key is to understand the infinite product (12.45) which becomes, using (12.46),

$$\begin{aligned} \Phi(\omega) &= \prod_{\ell=1}^{\infty} 2^{-1/2} \left( \frac{1 + e^{-j\omega/2^\ell}}{2} \right)^N R(e^{j\omega/2^\ell}) \\ &= \underbrace{\left( \prod_{\ell=1}^{\infty} \left( \frac{1 + e^{-j\omega/2^\ell}}{2} \right) \right)^N}_{A(\omega)} \underbrace{\prod_{\ell=1}^{\infty} 2^{-1/2} R(e^{j\omega/2^\ell})}_{B(\omega)}. \end{aligned} \quad (12.49)$$

Our goal is to see if  $\Phi(\omega)$  has a sufficiently fast decay for large  $\omega$ . We know from Chapter 3, (3.79a), that if  $|\Phi(\omega)|$  decays faster than  $1/|\omega|$  for large  $|\omega|$ , then  $\varphi(t)$  is bounded and continuous. Consider first the product

$$\prod_{\ell=1}^{\infty} \left( \frac{1 + e^{-j\omega/2^\ell}}{2} \right) \stackrel{(a)}{=} \prod_{\ell=1}^{\infty} 2^{-1/2} \frac{1}{\sqrt{2}} (1 + e^{-j\omega/2^\ell}) \stackrel{(b)}{=} e^{-j\omega/2} \frac{\sin(\omega/2)}{\omega/2},$$

where in (a) we extracted the Haar filter (12.1a), and (b) follows from (12.45) as well as the Haar case (12.7). The decay of this Fourier transform is of order  $O(1/|\omega|)$ , and thus,  $A(\omega)$  in (12.49) decays as  $O(1/|\omega|^N)$ .<sup>155</sup> So, as long as  $|B(\omega)|$  does not grow faster than  $|\omega|^{N-1-\epsilon}$ ,  $\epsilon > 0$ , the product (12.49) will decay fast enough to satisfy (3.79a), leading to a continuous scaling function  $\varphi(t)$ . We formalize this discussion in the following proposition, the proof of which is given in Solved Exercise 12.2:

<sup>155</sup>In time domain, it is the convolution of  $N$  box functions, or a  $B$  spline of order  $N - 1$  (see Chapter 5).

PROPOSITION 12.4 (SMOOTHNESS OF THE SCALING FUNCTION) With  $R(e^{j\omega})$  as in (12.46), if

$$B = \sup_{\omega \in [0, 2\pi]} |R(e^{j\omega})| < 2^{N-1/2}, \quad (12.50)$$

then, as  $J \rightarrow \infty$ , the iterated function  $\varphi^{(J)}(t)$  converges pointwise to a continuous function  $\varphi(t)$  with the Fourier transform

$$\Phi(\omega) = \prod_{\ell=1}^{\infty} 2^{-1/2} G(e^{j\omega/2^\ell}).$$

Condition (12.50) is sufficient, but not necessary: many filters fail the test but still lead to continuous limits (and more sophisticated tests can be used).

If we strengthened the bound to

$$B < 2^{N-k-1/2} \quad k \in \mathbb{N},$$

then  $\varphi(t)$  would be continuous and  $k$ -times differentiable (see Exercise TBD).

EXAMPLE 12.2 (SMOOTHNESS OF THE SCALING FUNCTION) We now test the continuity condition (12.50) on the two filters we have used most often.

The Haar filter

$$G(e^{j\omega}) = \frac{1}{\sqrt{2}}(1 + e^{-j\omega}) = \left( \frac{1 + e^{-j\omega}}{2} \right) \underbrace{\sqrt{2}}_{R(e^{j\omega})},$$

has  $N = 1$  zero at  $\omega = \pi$  and  $R(z) = \sqrt{2}$ . Thus,  $B = \sqrt{2}$ , which does not meet the inequality in (12.50). According to Proposition 12.4,  $\varphi(t)$  may or may not be continuous (and we know it is not).

The Daubechies filter (9.10)

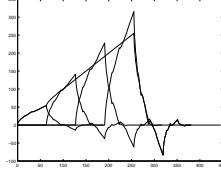
$$G(e^{j\omega}) = \left( \frac{1 + e^{-j\omega}}{2} \right)^2 \underbrace{\frac{1}{\sqrt{2}}(1 + \sqrt{3} + (1 - \sqrt{3})e^{-j\omega})}_{R(e^{j\omega})},$$

has  $N = 2$  zero at  $\omega = \pi$ . The supremum of  $|R(e^{j\omega})|$  is attained at  $\omega = \pi$ ,

$$B = \sup_{\omega \in [0, 2\pi]} |R(e^{j\omega})| = \sqrt{6} < 2^{3/2},$$

and thus, the scaling function  $\varphi(t)$  must be continuous.

**Reproduction of Polynomials** We have seen in Chapter 5 that splines of order  $N$  and their shifts can reproduce polynomials of degree up to  $N$ . Given that the scaling functions based on a filter having  $N$  zeros at  $\omega = \pi$  contain a *spline part* of



**Figure 12.24:** An example of the reproduction of polynomials by the scaling function and its shifts. The scaling function  $\varphi(t)$  is based on the Daubechies filter with two zeros at  $\pi$ , (9.10), and reproduces the linear function  $x(t) = t$  (on an interval because of the finite number of scaling functions used).

order  $(N - 1)$ , linear combinations of  $\{\varphi(t - n)\}_{n \in \mathbb{Z}}$  can reproduce polynomials of degree  $(N - 1)$ . We illustrate this property in Figure 12.24, where the Daubechies filter with two zeros at  $\pi$ , (9.10), reproduces the linear function  $x(t) = t$ . (We give the proof of this property later in the chapter.)

**Orthogonality to Integer Shifts** As we have seen in the Haar case already, the scaling function is orthogonal to its integer shifts, a property inherited from the underlying filter:

$$\langle \varphi(t), \varphi(t - n) \rangle_t = \delta_n. \quad (12.51)$$

Since  $\varphi(t)$  is defined through a limit and the inner product is continuous in both arguments, orthogonality (12.51) follows from the orthogonality of  $\varphi^{(J)}(t)$  and its integer shifts for any  $J$ :

$$\langle \varphi^{(J)}(t), \varphi^{(J)}(t - n) \rangle_t = \delta_n, \quad J \in \mathbb{Z}^+, \quad (12.52)$$

which follows, in turn, from the same property for the iterated filter  $g_n^{(J)}$  in (9.6a):

$$\begin{aligned} & \langle \varphi^{(J)}(t), \varphi^{(J)}(t - k) \rangle_t \\ & \stackrel{(a)}{=} \left\langle \sum_{n=0}^{L^{(J)}-1} g_n^{(J)} 2^{J/2} \varphi_h(2^J t - n), \sum_{m=0}^{L^{(J)}-1} g_m^{(J)} 2^{J/2} \varphi_h(2^J(t - k) - m) \right\rangle_t \\ & \stackrel{(b)}{=} \sum_{n=0}^{L^{(J)}-1} \sum_{m=0}^{L^{(J)}-1} g_n^{(J)} g_m^{(J)} \int_{-\infty}^{\infty} 2^J \varphi_h(2^J t - n) \varphi_h(2^J t - 2^J k - m) dt \\ & \stackrel{(c)}{=} \sum_{n=0}^{L^{(J)}-1} g_n^{(J)} g_{n-2^J k}^{(J)} = \langle g_n^{(J)}, g_{n-2^J k}^{(J)} \rangle_n \stackrel{(d)}{=} \delta_k, \end{aligned}$$

where (a) follows from (12.40); in (b) we took the sums and filter coefficients out of the inner product; (c) from the orthogonality of the Haar scaling functions; and (d) from the orthogonality of the filters themselves, (9.6a).

The orthogonality (12.51) at scale 0 has counterparts at other scales:

$$\langle \varphi(2^\ell t), \varphi(2^\ell t - n) \rangle_t = 2^{-\ell} \delta_n, \quad (12.53)$$

easily verified by changing the integration variable.

### 12.2.3 Wavelet Function and its Properties

The scaling function we have just seen is lowpass in nature (if the underlying filter  $g$  is lowpass in nature). Similarly to what we have done for the scaling function, we can construct a *wavelet function* (or, simply *wavelet*) that will be bandpass in nature (if the underlying filter  $h$  is highpass in nature).

We thus associate a piecewise-constant function  $\psi^{(J)}(t)$  to  $h_n^{(J)}$ , the impulse response of (12.37b), in such a way that  $\psi^{(J)}(t)$  is of finite length and of norm 1; we use the same arguments as before to determine the width and height of the piecewise segments, leading to

$$\psi^{(J)}(t) = 2^{J/2} h_n^{(J)} \quad \frac{n}{2^J} \leq t < \frac{n+1}{2^J}. \quad (12.54)$$

Unlike  $\varphi^{(J)}(t)$ , our new object of interest  $\psi^{(J)}(t)$  is a bandpass function. In particular, because  $H(e^{j\omega})|_{\omega=\pi} = 0$ , its Fourier transform  $\Psi(\omega)$  satisfies

$$\Psi(\omega)|_{\omega=0} = 0.$$

Again, we are interested in what happens when  $J \rightarrow \infty$ . Clearly, this involves an infinite product, but it is the same infinite product we studied for the convergence of  $\varphi^{(J)}(t)$  towards  $\varphi(t)$ . In short, we assume this question to be settled. The development parallels the one for the scaling function, with the important twist of consistently replacing the lowpass filter  $G(z^{2^{J-1}})$  by the highpass filter  $H(z^{2^{J-1}})$ . We do not repeat the details, but rather indicate the main points. Equation (12.44) becomes

$$\Psi^{(J)}(\omega) = \Phi_h^{(J)}(\omega) H(e^{j\omega/2}) \prod_{\ell=2}^J G(e^{j\omega/2^\ell}). \quad (12.55)$$

Similarly to the scaling function, we define the wavelet as the limit of  $\psi^{(J)}(t)$  or  $\Psi^{(J)}(\omega)$ , where we now assume that both are well defined and form a Fourier-transform pair.

**DEFINITION 12.5 (WAVELET)** Assuming the limit to exist, we define the wavelet in time and frequency domains to be

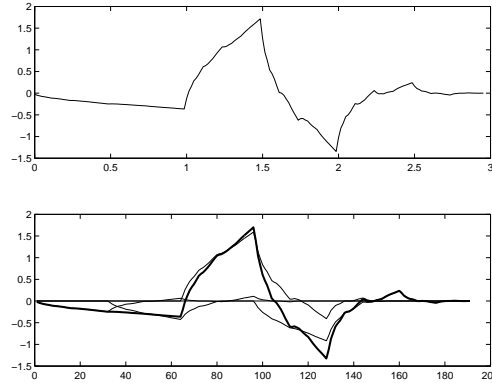
$$\psi(t) = \lim_{J \rightarrow \infty} \psi^{(J)}(t), \quad (12.56a)$$

$$\Psi(\omega) = \lim_{J \rightarrow \infty} \Psi^{(J)}(\omega). \quad (12.56b)$$

From (12.55) and using the steps leading to (12.45), we can write

$$\Psi(\omega) = 2^{-1/2} H(e^{j\omega/2}) \prod_{\ell=2}^{\infty} 2^{-1/2} G(e^{j\omega/2^\ell}). \quad (12.57)$$





**Figure 12.25:** Wavelet based on the Daubechies highpass filter (12.60). (a) Wavelet  $\psi(t)$  and (b) the two-scale equation for the wavelet.

**Two-Scale Equation** Similarly to (12.47), we can rewrite (12.57) as

$$\Psi(\omega) = 2^{-1/2} H(e^{j\omega/2}) \Phi(\omega/2). \quad (12.58)$$

Taking the inverse Fourier transform, we get a relation similar to (12.48), namely

$$\psi(t) = \sqrt{2} \sum_{n=0}^{L-1} h_n \varphi(2t - n), \quad (12.59)$$

the two-scale equation for the wavelet. From the support of  $\varphi(t)$  in (12.39), it also follows that  $\psi(t)$  has the same support on  $[0, L - 1]$ . To illustrate the two-scale relation and also show a wavelet, consider the following example.

**EXAMPLE 12.3 (WAVELET AND THE TWO-SCALE EQUATION)** Take the Daubechies lowpass filter (9.10) and construct its highpass via (7.24). It has a double zero at  $\omega = 0$ , and is given by:

$$H(z) = \frac{1}{4\sqrt{2}} \left[ (\sqrt{3} - 1) + (3 - \sqrt{3})z^{-1} - (3 + \sqrt{3})z^{-2} + (1 + \sqrt{3})z^{-3} \right]. \quad (12.60)$$

Figure 12.25 shows the wavelet  $\psi(t)$  and the two-scale equation.

**Smoothness** Since the wavelet is a finite linear combination of scaling functions and their shifts as in (12.59), the smoothness is inherited from the scaling function, as illustrated in Figure 12.25(a).

**Zero-Moment Property** We assumed that the lowpass filter  $G(e^{j\omega})$  had  $N$  zeros ( $N \geq 1$ ) at  $\omega = \pi$ . Using (7.24) and applying it to (12.46), we get

$$H(z) = e^{j(L-1)\omega} \left( \frac{1 - e^{j\omega}}{2} \right)^N R(e^{j(\omega+\pi)}). \quad (12.61)$$

It has therefore  $N$  zeros at  $\omega = 0$ . These  $N$  zeros carry over directly to  $\Psi(\omega)$  because of (12.58) and  $\Phi(\omega)|_{\omega=0} = 1$ . Because of this

$$\left. \frac{d^n X(\omega)}{d\omega^n} \right|_{\omega=0} = 0. \quad (12.62)$$

We can now use the moment property of the Fourier transform, (3.63a), to find the Fourier-transform pair of the above equation, leading to

$$\int_{-\infty}^{\infty} t^n \psi(t) dt = 0 \quad n = 0, 1, \dots, N-1. \quad (12.63)$$

In other words, if  $p(t)$  is a polynomial function of degree  $(N-1)$ , its inner product with the wavelet at any shift and/or scale will be 0:

$$\langle p(t), \psi(at - b) \rangle_t = 0 \quad \text{for all } a, b \in \mathbb{R}. \quad (12.64)$$

Remembering that  $\varphi(t)$  is able to reproduce polynomials up to degree  $(N-1)$ , it is a good role split for the two functions: wavelets annihilate polynomial functions while scaling functions reproduce them.

**Orthogonality to Integer Shifts** In our quest towards building orthonormal bases of wavelets, we will need that the wavelet is orthogonal to its integer shifts. The derivation is analogous to that for the scaling function; we thus skip it here, and instead just summarize this and other orthogonality conditions:

$$\langle \psi(2^\ell t), \psi(2^\ell t - n) \rangle_t = 2^{-\ell} \delta_n, \quad (12.65a)$$

$$\langle \varphi(2^\ell t), \psi(2^\ell t - n) \rangle_t = 0. \quad (12.65b)$$

### 12.2.4 Scaling Function and Wavelets from Biorthogonal Filter Banks

As we have already seen with filter banks, not all cases of interest are necessarily orthogonal. In Chapter 7, we designed biorthogonal filter banks to obtain symmetric/antisymmetric FIR filters. Similarly, with wavelets, except for the Haar case, there exist no orthonormal and compactly-supported wavelet bases that are symmetric/antisymmetric. Since symmetry is often a desirable feature, we need to relax orthonormality. We thus set the stage here for the biorthogonal wavelet series by briefly going through the necessary concepts.

To start, we assume a quadruple  $(h_n, g_n, \tilde{h}_n, \tilde{g}_n)$  of biorthogonal impulse responses satisfying the four biorthogonality relations (7.64a)–(7.64d). We further require that both lowpass filters have at least one zero at  $\omega = \pi$ , and more if possible:

$$G(e^{j\omega}) = \left( \frac{1 + e^{-j\omega}}{2} \right)^N R(e^{j\omega}), \quad \tilde{G}(e^{j\omega}) = \left( \frac{1 + e^{-j\omega}}{2} \right)^{\tilde{N}} \tilde{R}(e^{j\omega}). \quad (12.66)$$

## 12.2. Scaling Function and Wavelets from Orthogonal Filter Banks

819

Since the highpass filters are related to the lowpass ones by (7.75), the highpass filters  $H(e^{j\omega})$  and  $\tilde{H}(e^{j\omega})$  will have  $\tilde{N}$  and  $N$  zeros at  $\omega = 0$ , respectively. In the biorthogonal case, unlike in the orthonormal one, there is no implicit normalization, so we will assume that

$$G(e^{j\omega})|_{\omega=0} = \tilde{G}(e^{j\omega})|_{\omega=0} = \sqrt{2},$$

which can be enforced by normalizing  $H(e^{j\omega})$  and  $\tilde{H}(e^{j\omega})$  accordingly.

Analogously to the orthogonal case, the iterated filters are given by

$$G^{(J)}(z) = \prod_{\ell=0}^{J-1} G(z^{2^\ell}), \quad \tilde{G}^{(J)}(z) = \prod_{\ell=0}^{J-1} \tilde{G}(z^{2^\ell}),$$

and we define scaling functions in the Fourier domain as

$$\Phi(\omega) = \prod_{\ell=0}^{\infty} 2^{-1/2} G(e^{j\omega/2^\ell}), \quad \tilde{\Phi}(\omega) = \prod_{\ell=0}^{\infty} 2^{-1/2} \tilde{G}(e^{j\omega/2^\ell}). \quad (12.67)$$

In the sequel, we will concentrate on well-behaved cases only, that is, when the infinite products are well defined. Also, the iterated time-domain functions corresponding to  $G^{(J)}(z)$  and  $\tilde{G}^{(J)}(z)$  have well-defined limits  $\varphi(t)$  and  $\tilde{\varphi}(t)$ , respectively, related to (12.67) by Fourier transform.

The two-scale relations follow similarly to the orthogonal case:

$$\Phi(\omega) = 2^{-1/2} G(e^{j\omega/2}) \Phi(\omega/2), \quad (12.68a)$$

$$\varphi(t) = \sqrt{2} \sum_{n=0}^{L-1} g_n \varphi(2t - n), \quad (12.68b)$$

as well as

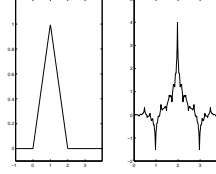
$$\tilde{\Phi}(\omega) = 2^{-1/2} \tilde{G}(e^{j\omega/2}) \tilde{\Phi}(\omega/2), \quad (12.69a)$$

$$\tilde{\varphi}(t) = \sqrt{2} \sum_{n=0}^{L-1} g_n \tilde{\varphi}(2t - n). \quad (12.69b)$$

**EXAMPLE 12.4 (SCALING FUNCTION AND WAVELETS FROM LINEAR  $B$ -SPLINES)**  
Choose as the lowpass filter

$$G(e^{j\omega}) = \sqrt{2} e^{j\omega} \left( \frac{1 + e^{-j\omega}}{2} \right)^2 = \frac{1}{2\sqrt{2}} (e^{j\omega} + 2 + e^{-j\omega}),$$

which has a double zero at  $\omega = 0$  and satisfies the normalization  $G(e^{j\omega})|_{\omega=0} = \sqrt{2}$ . Then, using (12.67), we compute  $\Phi(\omega)$  to be (3.49f), that is, the Fourier transform of the hat function (3.49a), or linear  $B$ -spline. This is because  $G(z)$  is (up to a normalization and shift) the convolution of the Haar filter with itself.



**Figure 12.26:** The hat function and its dual. (a)  $\varphi(t)$  from the iteration of  $\frac{1}{2\sqrt{2}}[1, 2, 1]$ . (b)  $\tilde{\varphi}(t)$  from the iteration of  $\frac{1}{4\sqrt{2}}[-1, 2, 6, 2, -1]$ .

Thus, the limit of the iterated filter is the convolution of the box function with itself, the result being shifted to be centered at the origin.

We now search for a biorthogonal scaling function  $\tilde{\varphi}(t)$  by finding first a (nonunique) biorthogonal lowpass filter  $\tilde{G}(z)$  satisfying (7.66). Besides the trivial solution  $\tilde{G}(z) = 1$ , the following is a solution as well:

$$\tilde{G}(z) = \frac{1}{4\sqrt{2}}(1+z)(1+z^{-1})(-z+4-z^{-1}) = \frac{1}{4\sqrt{2}}(-z^2+2z+6+2z^{-1}-z^{-2}),$$

obtained as one possible factorization of  $C(z)$  from Example 7.4. The resulting dual scaling function  $\tilde{\varphi}(t)$  looks quite irregular (see Figure 12.26). We could, instead, look for a  $\tilde{G}(z)$  with more zeros at  $\omega = \pi$  to obtain a smoother dual scaling function. For example, choose

$$\tilde{G}(z) = \frac{1}{64\sqrt{2}}(1+z)^2(1+z^{-1})^2(3z^2-18z+38-18z^{-1}+3z^{-2}),$$

leading to quite a different  $\tilde{\varphi}(t)$  (see Figure 12.27).

Choosing the highpass filters as in (7.75),

$$H(z) = z\tilde{G}(-z^{-1}), \quad \tilde{H}(z) = z^{-1}G(-z),$$

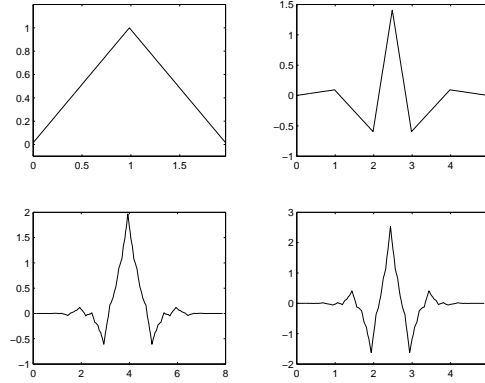
with only a minimal shift, since the lowpass filters are centered around the origin and symmetric, we get all four functions as in Figure 12.27.

## 12.3 Wavelet Series

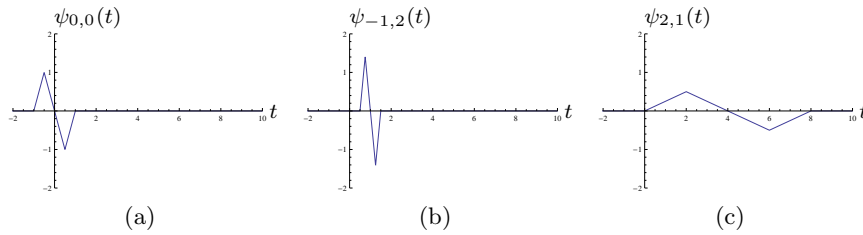
So far, we have considered only a single scale with the two functions  $\varphi(t)$  and  $\psi(t)$ . Yet, as for the Haar case in Section 12.1, multiple scales are already lurking in the background through the two-scale equations (12.48),(12.59). And just like in the DWT in Chapter 9, the real action appears when all scales are considered as we have already seen with the Haar wavelet series.

## 12.3. Wavelet Series

821



**Figure 12.27:** Biorthogonal linear spline basis. (a) The linear  $B$ -spline is the hat function  $\varphi(t)$ . (b) The linear  $B$ -spline wavelet  $\psi(t)$ . (c) The dual scaling function  $\tilde{\varphi}(t)$ . (d) The dual wavelet  $\tilde{\psi}(t)$ .



**Figure 12.28:** Example wavelets. (a) The prototype wavelet  $\psi(t) = \psi_{0,0}(t)$ ; (b)  $\psi_{-1,2}(t)$ ; (c)  $\psi_{2,1}(t)$ .

## 12.3.1 Definition of the Wavelet Series

We thus recall

$$\psi_{\ell,k}(t) = 2^{-\ell/2} \psi(2^{-\ell}t - k) = \frac{1}{2^{\ell/2}} \psi\left(\frac{t - 2^\ell k}{2^\ell}\right), \quad (12.70a)$$

$$\varphi_{\ell,k}(t) = 2^{-\ell/2} \varphi(2^{-\ell}t - k) = \frac{1}{2^{\ell/2}} \varphi\left(\frac{t - 2^\ell k}{2^\ell}\right), \quad (12.70b)$$

for  $\ell, k \in \mathbb{Z}$ , with the understanding that the basic scaling function  $\varphi(t)$  and the wavelet  $\psi(t)$  are no longer Haar, but can be more general. As before, for  $\ell = 0$ , we have the usual scaling function and wavelet and their integer shifts; for  $\ell > 0$ , the functions are stretched by a power of 2, and the shifts are proportionally increased; and for  $\ell < 0$ , the functions are compressed by a power of 2, with appropriately reduced shifts. Both the scaling function and the wavelet are of unit norm, and that at all scales (a few examples are given in Figure 12.28).

**Two-Scale Equations at Nonconsecutive Scales** Since we want to deal with multiple scales (not just two), we extend the two-scale equations for  $\varphi(t)$  and  $\psi(t)$  across arbitrary scales that are powers of 2:

$$\begin{aligned}\Phi(\omega) &\stackrel{(a)}{=} 2^{-1/2} G(e^{j\omega/2}) \Phi(\omega/2), \\ &\stackrel{(b)}{=} 2^{-1} G(e^{j\omega/2}) G(e^{j\omega/4}) \Phi(\omega/4), \\ &\stackrel{(c)}{=} 2^{-1} G^{(2)}(e^{j\omega/4}) \Phi(\omega/4), \\ &\vdots \\ &= 2^{-k/2} G^{(k)}(e^{j\omega/2^k}) \Phi(\omega/2^k),\end{aligned}\tag{12.71a}$$

$$\varphi(t) = 2^{k/2} \sum_{n=0}^{L-1} g_n^{(k)} \varphi(2^k t - n),\tag{12.71b}$$

for  $k \in \mathbb{Z}^+$ , where both (a) and (b) follow from the two-scale equation in the Fourier domain, (12.47); (c) from the expression for the equivalent filter, (12.37a); and (d) by repeatedly applying the same (see Exercise 12.3). The last expression is obtained by applying the inverse DTFT to (12.71a).

Using an analogous derivation for the wavelet, we get

$$\Psi(\omega) = 2^{-k/2} H^{(k)}(e^{j\omega/2^k}) \Phi(\omega/2^k),\tag{12.72a}$$

$$\psi(t) = 2^{k/2} \sum_{n=0}^{L-1} h_n^{(k)} \varphi(2^k t - n),\tag{12.72b}$$

for  $k = 2, 3, \dots$ . The attractiveness of the above expressions lies in their ability to express any  $\varphi_{\ell,k}(t)$ ,  $\psi_{\ell,k}(t)$ , in terms of a linear combination of an appropriately scaled  $\varphi(t)$ , where the linear combination is given by the coefficients of an equivalent filter  $g_n^{(k)}$  or  $h_n^{(k)}$ . We are now ready for the main result of this chapter:

**THEOREM 12.6 (ORTHONORMAL BASIS FOR  $\mathcal{L}^2(\mathbb{R})$ )** The continuous-time wavelet  $\psi(t)$  satisfying (12.59) and its shifts and scales,

$$\{\psi_{\ell,k}(t)\} = \left\{ \frac{1}{\sqrt{2^\ell}} \psi\left(\frac{t - 2^\ell k}{2^\ell}\right) \right\}, \quad \ell, k \in \mathbb{Z},\tag{12.73}$$

form an orthonormal basis for the space of square-integrable functions,  $\mathcal{L}^2(\mathbb{R})$ .

*Proof.* To prove the theorem, we must prove that (i)  $\{\psi_{\ell,k}(t)\}_{\ell, k \in \mathbb{Z}}$  is an orthonormal set and (ii) it is complete. The good news is that most of the hard work has already been done while studying the DWT in Theorem 9.2, Chapter 9.

(i) We have already shown that the wavelets and their shifts are orthonormal at a

single scale, (12.65a), and need to show the same across scales:

$$\begin{aligned}
 \langle \psi_{\ell,k}(t), \psi_{m,n}(t) \rangle_t &\stackrel{(a)}{=} 2^{-\ell} \langle \psi_{0,k}(\tau), \psi_{-i,n}(\tau) \rangle_{\tau}, \\
 &\stackrel{(b)}{=} 2^{-\ell} \langle 2^{i/2} \sum_n h_n^{(i)} \varphi_{-i,2^i k+n}(\tau), \psi_{-i,n}(\tau) \rangle_{\tau}, \\
 &\stackrel{(c)}{=} 2^{-\ell+i/2} \sum_n h_n^{(i)} \langle \varphi_{-i,2^i k+n}(\tau), \psi_{-i,n}(\tau) \rangle_{\tau} = 0,
 \end{aligned}$$

where (a) follows from assuming (without loss of generality)  $\ell = m + i$ ,  $i > 0$ , and change of variable  $t = 2^\ell \tau$ ; (b) from two-scale equation for the wavelet (12.72b); and (c) from the linearity of the inner product as well as orthogonality of the wavelet and scaling function (12.65a).

- (ii) The proof of completeness is more involved, and thus, we show it only for the Haar case. *Further Reading* gives pointers to texts with the full proof. Consider a unit-norm function  $x(t)$  such that  $x(t) = 0$  for  $t < 0$  with finite length at most  $2^J$  for some  $J \in \mathbb{Z}$ .<sup>156</sup> We approximate  $x(t)$  by a piecewise-constant approximation at scale  $\ell$ , (where  $\ell \ll J$ ), or

$$\begin{aligned}
 x^{(\ell)}(t) &= 2^{-\ell} \int_{2^\ell k}^{2^\ell(k+1)} x(\tau) d\tau, \quad 2^\ell k \leq t < 2^\ell(k+1), \\
 &\stackrel{(a)}{=} \sum_{k \in \mathbb{Z}} \left( \int_{\tau \in \mathbb{R}} x(\tau) \varphi_{\ell,k}(\tau) d\tau \right) \varphi_{\ell,k}(t), \\
 &\stackrel{(b)}{=} \sum_{k \in \mathbb{Z}} \langle x, \varphi_{\ell,k} \rangle \varphi_{\ell,k}(t) \stackrel{(c)}{=} \sum_{k \in \mathbb{Z}} \alpha_k^{(\ell)} \varphi_{\ell,k}(t), \tag{12.74a}
 \end{aligned}$$

where (a) follows from (12.16b); (b) from the definition of the inner product; and in (c) we introduced  $\alpha_k^{(\ell)} = \langle x, \varphi_{\ell,k} \rangle$ .

Because of the finite-length assumption of  $x(t)$ , the sequence  $\alpha_k^{(\ell)}$  is also of finite length (of order  $2^{J-\ell}$ ). Since  $x(t)$  is of norm 1 and the approximation in (12.74a) is a projection,  $\|\alpha_k^{(\ell)}\| \leq 1$ . Thus, we can apply Theorem 9.2 and represent the sequence  $\alpha_k^{(\ell)}$  by discrete Haar wavelets only

$$\alpha_k^{(\ell)} = \sum_{n \in \mathbb{Z}} \sum_{i \in \mathbb{Z}^+} \beta_n^{(i)} h_{k-2^i n}^{(i)}.$$

Since the expression (12.74a) is the piecewise-constant interpolation of the sequence  $\alpha_k^{(\ell)}$ , together with proper scaling and normalization, by linearity, we can apply this interpolation to the discrete Haar wavelets used to represent  $\alpha_k^{(\ell)}$ , which

<sup>156</sup> Both of these restrictions are inconsequential; the former because a general function can be decomposed into a function nonzero on  $t > 0$  and  $t \leq 0$ , the latter because the fraction of the energy of  $x(t)$  outside of the interval under consideration can be made arbitrarily small by making  $J$  arbitrarily large.

leads to a continuous-time Haar wavelet representation of  $x^{(\ell)}(t)$ :

$$\begin{aligned} x^{(\ell)}(t) &= \sum_{k \in \mathbb{Z}} \alpha_k^{(\ell)} \varphi_{\ell,k}(t), \\ &= \sum_{k \in \mathbb{Z}} \sum_{n \in \mathbb{Z}} \sum_{i \in \mathbb{Z}^+} \beta_n^{(i)} h_{k-2^i n}^{(i)} \varphi_{\ell,k}(t), \\ &= \sum_{n \in \mathbb{Z}} \sum_{i \in \mathbb{Z}^+} \beta_n^{(i)} \psi_{\ell+i, n 2^i}(t). \end{aligned}$$

This last statement follows from  $h^{(i)}$  being of length  $2^i$ . Thus, for a fixed  $n$  and  $i$ ,  $\sum_{k \in \mathbb{Z}} h_{k-2^i n}^{(i)} \varphi_{\ell,k}(t)$  will equal the Haar wavelet of length  $2^i 2^\ell = 2^{i+\ell}$  at shift  $n 2^i$ . Again by Theorem 9.2, this representation is exact.

What remains to be shown is that  $x^{(\ell)}(t)$  can be made arbitrarily close, in  $\mathcal{L}^2$  norm, to  $x(t)$ . This is achieved by letting  $\ell \rightarrow -\infty$  and using the fact that piecewise-constant functions are dense in  $\mathcal{L}^2(\mathbb{R})$ ,<sup>157</sup> we get

$$\lim_{i \rightarrow -\infty} \|x(t) - x^{(\ell)}(t)\| = 0.$$

TBD: Might be expanded.

The proof once more shows the intimate relation between the DWT from Chapter 9 and the wavelet series from this chapter.

**Definition** We can now formally define the wavelet series:

**DEFINITION 12.7 (WAVELET SERIES)** The wavelet series of a function  $x(t)$  is a function of  $\ell, k \in \mathbb{Z}$  given by

$$\beta_k^{(\ell)} = \langle x, \psi_{\ell,k} \rangle = \int_{-\infty}^{\infty} x(t) \psi_{\ell,k}(t) dt, \quad \ell, k \in \mathbb{Z}, \quad (12.75a)$$

with  $\psi_{\ell,k}(t)$  the prototype wavelet. The inverse wavelet series is given by

$$x(t) = \sum_{\ell \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} \beta_k^{(\ell)} \psi_{\ell,k}(t). \quad (12.75b)$$

In the above,  $\beta_k^{(\ell)}$  are the *wavelet coefficients*.

To denote such a wavelet series pair, we write:

$$x(t) \xleftrightarrow{\text{WS}} \beta_k^{(\ell)}.$$

We derived such bases already and we will see other constructions when we talk about multiresolution analysis.

<sup>157</sup> That is, any  $\mathcal{L}^2$  function can be approximated arbitrarily closely by a piecewise-constant function over intervals that tend to 0. This is a standard result but technical, and thus we just use it without proof.



### 12.3.2 Properties of the Wavelet Series

We now consider some of the properties of the wavelet series. Many follow from the properties of the wavelet (Section 12.2.3) or of the DWT (Chapter 9), and thus our treatment will be brief.

**Linearity** The wavelet series operator is a linear operator, or,

$$a x(t) + b y(t) \xleftrightarrow{\text{WS}} a \beta_k^{(\ell)} + b \beta_k^{(\ell)}. \quad (12.76)$$

**Shift in Time** A shift in time by  $2^m n$ ,  $m, n \in \mathbb{Z}$ , results in

$$x(t - 2^m n) \xleftrightarrow{\text{WS}} \beta_{k-2^m n}^{(\ell)}, \quad \ell \leq m. \quad (12.77)$$

This is a restrictive condition as it holds only for scales smaller than  $m$ . In other words, only a function  $x(t)$  that has a scale-limited expansion, that is, it can be written as

$$x(t) = \sum_{\ell=-\infty}^m \sum_{k \in \mathbb{Z}} \beta_k^{(\ell)} \psi_{\ell,k}(t),$$

will possess the shift-in-time property for all (of its existing) scales. This is a counterpart to the shift-in-time property of the DWT, (9.17), and the fact that the DWT is periodically shift variant.

**Scaling in Time** Scaling in time by  $2^{-m}$ ,  $m \in \mathbb{Z}$ , results in

$$x(2^{-m} t) \xleftrightarrow{\text{WS}} 2^{m/2} \beta_k^{(\ell-m)}. \quad (12.78)$$

**Parseval's Equality** The wavelet series operator is a unitary operator and thus preserves the Euclidean norm (see (1.51)):

$$\|x\|^2 = \int_{-\infty}^{\infty} |x(t)|^2 dt = \sum_{\ell \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} |\beta_k^{(\ell)}|^2. \quad (12.79)$$

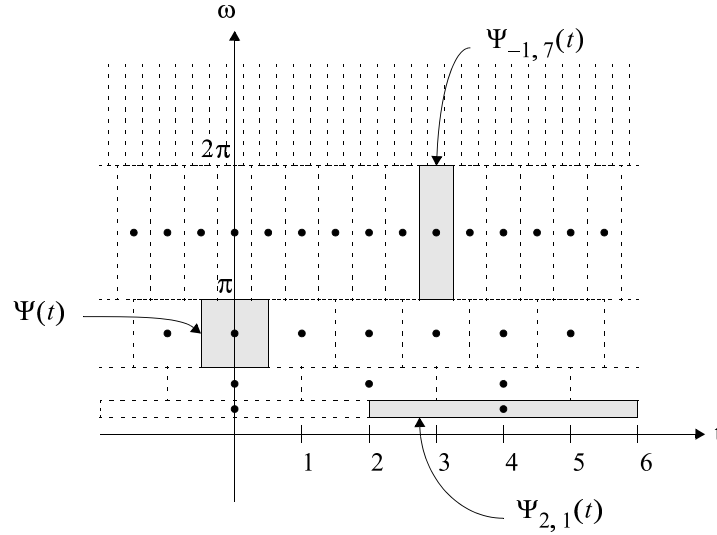
**Time-Frequency Localization** Assume that the wavelet  $\psi(t)$  is centered around  $t = 0$  in time and  $\omega = 3\pi/4$  in frequency (that is, it is a bandpass filter with the support of approximately  $[\pi/2, \pi]$ ). Then, from (12.70a),  $\psi_{\ell,0}(t)$  is centered around  $\omega = 2^{-\ell}(3\pi/4)$  in frequency (see Figure 12.29).

With our assumption of  $g$  being a causal FIR filter of length  $L$ , the support in time of the wavelets is easy to characterize. Since the support of  $\psi(t)$  is  $[0, L-1]$ ,

$$\text{support}(\psi_{\ell,k}(t)) \subseteq [2^\ell k, 2^\ell(k+L-1)). \quad (12.80)$$

Because of the FIR assumption, the frequency localization is less precise (no compact support in frequency), but the center frequency is around  $2^{-\ell}(3\pi/4)$  and the passband is mostly in an octave band,

$$\text{support}(\Psi_{\ell,k}(\omega)) \sim [2^{-\ell} \frac{\pi}{2}, 2^{-\ell} \pi]. \quad (12.81)$$



**Figure 12.29:** Time-frequency localization of wavelet basis functions. Three wavelets are highlighted: at scale  $\ell = 0$ ,  $\psi(t)$ ; at scale  $\ell = -1$ , a higher-frequency wavelet  $\psi_{-1,7}(t)$ ; and at scale  $\ell = 2$ , a lower-frequency wavelet  $\psi_{2,1}(t)$ . These are centered along the dyadic sampling grid  $[2^\ell k - 2^{-\ell}(3\pi/4)]$ , for  $\ell, k \in \mathbb{Z}$ .

**Characterization of Singularities** As we have seen with the example of Haar wavelet series (see Figure 12.8), one of the powerful features of the wavelet series is its ability to characterize both the position and type of singularities present in a function.

Consider a function with the simplest singularity, a Dirac delta function at a location  $t_0$ , that is,  $x(t) = \delta(t - t_0)$ . At scale  $\ell$ , only wavelets having their support (12.80) straddling  $t_0$  will produce nonzero coefficients,

$$\beta_k^{(\ell)} \neq 0 \quad \text{for} \quad \lfloor t_0/2^\ell \rfloor - L < k \leq \lfloor t_0/2^\ell \rfloor. \quad (12.82)$$

Thus, there are  $L$  nonzero coefficients at each scale. These coefficients correspond to a region of size  $2^\ell(L-1)$  around  $t_0$ , or, as  $\ell \rightarrow -\infty$ , they focus arbitrarily closely on the singularity. What about the size of the coefficients at scale  $\ell$ ? The inner product of the wavelet with a Dirac delta function simply picks out a value of the wavelet. Because of the scaling factor  $2^{-\ell/2}$  in (12.70a), the nonzero coefficients will be of order

$$|\beta_k^{(\ell)}| \sim O(2^{-\ell/2}) \quad (12.83)$$

for the range of  $k$  in (12.82). That is, as  $\ell \rightarrow -\infty$ , the nonzero wavelet series coefficients zoom in onto the discontinuity, and they grow at a specific rate given by (12.83). An example for the Haar wavelet was shown in Figure 12.8.

Generalizing the Dirac delta function singularity, a function is said to have an  $n$ th-order singularity at  $t_0$  when its  $n$ th-order derivative has a Dirac delta function

component at  $t_0$ . The scaling (12.83) for a zeroth-order singularity is an example of the following result:

**PROPOSITION 12.8 (SCALING BEHAVIOR AROUND SINGULARITIES)** Given a wavelet  $\psi(t)$  with  $N$  zero moments, around a singularity of order  $n$ ,  $0 \leq n \leq N$ , the wavelet series coefficients  $\beta_k^{(\ell)}$  behave as

$$\left| \beta_k^{(\ell)} \right| \sim O(2^{\ell(n-1/2)}), \quad \ell \rightarrow -\infty. \quad (12.84)$$

*Proof.* We have analyzed  $n = 0$  earlier. We now give a proof for  $n = 1$ ; generalizing to  $n > 1$  is the topic of Exercise 12.3.

Assume the wavelet has at least one zero moment,  $N \geq 1$ . A function with a first-order singularity at  $t_0$  looks like a Heaviside function (3.8) locally (at  $t_0$ ). We can reduce the analysis to  $n = 0$  by considering the derivative  $x'(t)$ , which is a Dirac delta function at  $t_0$ . We use the fact that  $\psi$  has at least one zero moment and is of finite support. Then, as in (12.34), using integration by parts,

$$\begin{aligned} \langle x(t), \psi(t) \rangle_t &= \int_{-\infty}^{\infty} \psi(t)x(t) dt = - \int_{-\infty}^{\infty} \theta(t)x'(t) dt \\ &= - \langle x'(t), \theta(t) \rangle_t \\ \langle x(t), \psi_{\ell,k}(t) \rangle_t &= - \langle x'(t), \theta_{\ell,k}(t) \rangle_t, \end{aligned}$$

where  $\theta(t) = \int_{-\infty}^t \psi(\tau) d\tau$  is the primitive of  $\psi(t)$ ,  $\theta_{\ell,k}(t)$  is the primitive of  $\psi_{\ell,k}(t)$ , and  $x'(t)$  is the derivative of  $x(t)$ . Because  $\psi(t)$  has at least one zero at  $\omega = 0$  and is of finite support, its primitive is well defined and also of finite support. The key is now the scaling behavior of  $\theta_{\ell,k}(t)$  with respect to  $\theta(t)$ . Evaluating

$$\theta_{\ell,k}(t) = \int_{-\infty}^t 2^{-\ell/2} \psi(2^{-\ell}\tau - k) d\tau = 2^{\ell/2} \int_{-\infty}^{2^{-\ell}t - k} \psi(t') dt' = 2^{\ell/2} \theta(2^{-\ell}t - k),$$

we see that this scaling is given by  $2^{\ell/2}$ . Therefore, the wavelet coefficients scale as

$$\begin{aligned} \left| \beta_k^{(\ell)} \right| &= |\langle x(t), \psi_{\ell,k}(t) \rangle_t| = |-\langle x'(t), \theta_{\ell,k}(t) \rangle_t| \\ &\sim 2^{\ell/2} |\langle \delta(t - t_0), \theta(2^{-\ell}t - k) \rangle_t| \sim O(2^{\ell/2}), \end{aligned} \quad (12.85)$$

at fine scales and close to  $t_0$ .

**Zero-Moment Property** When the lowpass filter  $g$  has  $N$  zeros at  $\omega = \pi$ , we verified that  $\psi(t)$  has  $N$  zero moments (12.63). This property carries over to all scaled versions of  $\psi(t)$ , and thus, for any polynomial function  $p(t)$  of degree smaller than  $N$ ,

$$\beta_k^{(\ell)} = \langle p(t), \psi_{\ell,k}(t) \rangle_t = 0.$$

This allows us to prove the following result:

**PROPOSITION 12.9 (DECAY OF WAVELET SERIES COEFFICIENTS FOR  $x \in \mathbb{C}^N$ )**  
 For a function  $x(t)$  with  $N$  continuous and bounded derivations, that is,  $x \in C^N$ , the wavelet series coefficients decay as

$$\left| \beta_k^{(\ell)} \right| \leq \alpha 2^{mN}$$

for some constant  $\alpha > 0$  and  $m \rightarrow -\infty$ .

*Proof.* Consider the Taylor series expansion of  $x(t)$  around some point  $t_0$ . Since  $x(t)$  has  $N$  continuous derivatives,

$$\begin{aligned} x(t_0 + \epsilon) &= x(t_0) + \frac{x'(t_0)}{1!}\epsilon + \frac{x''(t_0)}{2!}\epsilon^2 + \cdots + \frac{x^{(N-1)}(t_0)}{(N-1)!}\epsilon^{N-1} + R_N(\epsilon), \\ &= p(t) + R_N(\epsilon), \end{aligned}$$

where

$$|R_N(\epsilon)| \leq \frac{\epsilon^N}{N!} \sup_{t_0 \leq t \leq t_0 + \epsilon} |x^{(N)}(t)|,$$

and we view it as a polynomial  $p(t)$  of degree  $(N-1)$  and a remainder  $R_N(\epsilon)$ . Because of the zero-moment property of the wavelet,

$$\left| \beta_k^{(\ell)} \right| = |\langle x(t), \psi_{m,n}(t) \rangle| = |\langle p(t) + R_N(\epsilon), \psi_{m,n}(t) \rangle| = |\langle R_N(\epsilon), \psi_{m,n}(t) \rangle|,$$

that is, the inner product with the polynomial term is zero, and only the remainder matters. To minimize the upper bound on  $|\langle R_N(\epsilon), \psi_{m,n} \rangle|$ , we want  $t_0$  close to the center of the wavelet. Since the spacing of the sampling grid at scale  $\ell$  is  $2^\ell$ , we see that  $\epsilon$  is at most  $2^\ell$  and thus  $|\langle R_N(\epsilon), \psi_{\ell,k} \rangle|$  has an upper bound of order  $2^{\ell N}$ .

A stronger result, in which  $N$  is replaced by  $N+1/2$ , follows from Proposition 12.17 in the context of the continuous wavelet transform.

### 12.3.3 Multiresolution Analysis

We have already introduced the concept of multiresolution analysis with the Haar scaling function and wavelet in Section 12.1. As opposed to having a discrete-time filter and constructing a continuous-time basis from it, multiresolution analysis does the opposite: it starts from the multiresolution spaces to build the wavelet series. For example, we saw that the continuous-time wavelet basis generated a partition of  $\mathcal{L}^2(\mathbb{R})$  into a sequence of nested spaces

$$\dots \subset V^{(2)} \subset V^{(1)} \subset V^{(0)} \subset V^{(-1)} \subset V^{(-2)} \subset \dots,$$

and that these spaces were all scaled copies of each other, that is,  $V^{(\ell)}$  is  $V^{(0)}$  scaled by  $2^\ell$ . We will turn the question around and ask: assuming we have a sequence of nested and scaled spaces as above, does it generate a discrete-time filter bank? The answer is yes; the framework is multiresolution analysis we have seen in the Haar

case. We present it shortly in its more general form, starting with the axiomatic definition and followed by examples.

The embedded spaces above are very natural for piecewise-polynomial functions over uniform intervals of length  $2^\ell$ . For example, the Haar case leads to piecewise-constant functions. The next higher order is for piecewise-linear functions, and so on. The natural bases for such spaces are  $B$ -splines we discussed in Chapter 5; these are not orthonormal bases, requiring the use of orthogonalization methods.

**Axioms of Multiresolution Analysis** We now summarize the fundamental characteristics of the spaces and basis functions seen in the Haar case. These are also the axioms of multiresolution analysis.

- (i) *Embedding*: We work with a sequence of embedded spaces

$$\dots \subset V^{(2)} \subset V^{(1)} \subset V^{(0)} \subset V^{(-1)} \subset V^{(-2)} \subset \dots, \quad (12.86a)$$

where  $V^{(\ell)}$  is the space of piecewise-constant functions over  $[2^\ell k, 2^\ell(k+1))_{k \in \mathbb{Z}}$  with finite  $\mathcal{L}^2$  norm. We call the  $V^{(\ell)}$ s *successive approximation spaces*, since as  $\ell \rightarrow -\infty$ , we get finer and finer approximations.

- (ii) *Upward Completeness*: Since piecewise-constant functions over arbitrarily-short intervals are dense in  $\mathcal{L}^2$  (see Footnote 157),

$$\lim_{\ell \rightarrow -\infty} V^{(\ell)} = \overline{\bigcup_{\ell \in \mathbb{Z}} V^{(\ell)}} = \mathcal{L}^2(\mathbb{R}). \quad (12.86b)$$

- (iii) *Downward Completeness*: As  $\ell \rightarrow \infty$ , we get coarser and coarser approximations. Given a function  $x(t) \in \mathcal{L}^2(\mathbb{R})$ , its projection onto  $V^{(\ell)}$  tends to zero as  $\ell \rightarrow \infty$ , since we lose all the details. More formally,

$$\bigcap_{\ell \in \mathbb{Z}} V^{(\ell)} = \{0\}. \quad (12.86c)$$

- (iv) *Scale Invariance*: The spaces  $V^{(\ell)}$  are just scaled versions of each other,

$$x(t) \in V^{(\ell)} \Leftrightarrow x(2^m t) \in V^{(\ell-m)}. \quad (12.86d)$$

- (v) *Shift Invariance*: Because  $x(t)$  is a piecewise-constant function over intervals  $[2^\ell k, 2^\ell(k+1))$ , it is invariant to shifts by multiples of  $2^\ell$ ,

$$x(t) \in V^{(\ell)} \Leftrightarrow x(t - 2^\ell k) \in V^{(\ell)}. \quad (12.86e)$$

- (vi) *Existence of a Basis*: There exists  $\varphi(t) \in V^{(0)}$  such that

$$\{\varphi(t - k)\}_{k \in \mathbb{Z}} \quad (12.86f)$$

is a basis for  $V^{(0)}$ .

The above six characteristics, which naturally generalize the Haar multiresolution analysis, are the defining characteristics of a broad class of wavelet systems.

**DEFINITION 12.10 (MULTIRESOLUTION ANALYSIS)** A sequence  $\{V^{(\ell)}\}_{m \in \mathbb{Z}}$  of subspaces of  $\mathcal{L}^2(\mathbb{R})$  satisfying (12.86a)–(12.86f) is called a multiresolution analysis. The spaces  $V^{(\ell)}$  are called the successive approximation spaces, while the spaces  $W^{(\ell)}$ , defined as the orthogonal complements of  $V^{(\ell)}$  in  $V^{(\ell-1)}$ , that is,

$$V^{(\ell-1)} = V^{(\ell)} \oplus W^{(\ell)}, \quad (12.87)$$

are called the successive detail spaces.

**Definition** For simplicity, we will assume the basis in (12.86f) to be orthonormal; we cover the general case in Solved Exercise 12.3.

The two-scale equation (12.48) follows naturally from the scale-invariance axiom ((iv)). What can we say about the coefficients  $g_n$ ? Evaluate

$$\begin{aligned} \delta_k &\stackrel{(a)}{=} \langle \varphi(t), \varphi(t-k) \rangle_t \stackrel{(b)}{=} 2 \sum_{n \in \mathbb{Z}} \sum_{m \in \mathbb{Z}} g_n g_m \langle \varphi(2t-n), \varphi(2t-2k-m) \rangle_t \\ &\stackrel{(c)}{=} \sum_{n \in \mathbb{Z}} g_n g_{n-2k}, \end{aligned}$$

where (a) is true by assumption; in (b) we substituted the two-scale equation 12.48 for both  $\varphi(t)$  and  $\varphi(t-k)$ ; and (c) follows from  $\langle \varphi(2t-n), \varphi(2t-2k-m) \rangle_t = 0$  except for  $n = 2k + m$  when it is 1/2. We thus conclude that the sequence  $g_n$  corresponds to an orthogonal filter (7.13). Assuming that the Fourier transform  $\Phi(\omega)$  of  $\varphi(t)$  is continuous and satisfies<sup>158</sup>

$$|\Phi(0)| = 1,$$

it follows from the two-scale equation in the Fourier domain that

$$|G(1)| = \sqrt{2},$$

making  $g_n$  a lowpass sequence. Assume it to be of finite length  $L$  and derive the equivalent highpass filter using (7.24). Defining the wavelet as in (12.59), we have:

**PROPOSITION 12.11** The wavelet given by (12.59) satisfies

$$\begin{aligned} \langle \psi(t), \psi(t-n) \rangle_t &= \delta_n, \\ \langle \psi(t), \varphi(t-n) \rangle_t &= 0, \end{aligned}$$

and  $W^{(0)} = \text{span}(\{\psi(t-n)\}_{n \in \mathbb{Z}})$  is the orthogonal complement of  $V^{(0)}$  in  $V^{(-1)}$ ,

$$V^{(-1)} = V^{(0)} \oplus W^{(0)}. \quad (12.88)$$

<sup>158</sup>If  $\varphi(t)$  is integrable, this follows from upward completeness (12.86b)) for example.

We do not prove the proposition but rather just discuss the outline of a proof. The orthogonality relations follow from the orthogonality of the sequences  $g_n$  and  $h_n$  by using the two-scale equations (12.48) and (12.59). That  $\{\psi(t - n)\}_{n \in \mathbb{Z}}$  is an orthonormal basis for  $W^{(0)}$  requires checking completeness and is more technical. By construction, and in parallel to (12.86d),  $W^{(\ell)}$  are just scaled versions of each other,

$$x(t) \in W^{(\ell)} \Leftrightarrow x(2^m t) \in W^{(\ell-m)}. \quad (12.89)$$

Putting all the pieces above together, we have:

**THEOREM 12.12 (WAVELET BASIS FOR  $\mathcal{L}^2(\mathbb{R})$ )** Given a multiresolution analysis of  $\mathcal{L}^2(\mathbb{R})$  from Definition 12.10, the family

$$\psi_{\ell,k}(t) = \frac{1}{2^{\ell/2}} \psi\left(\frac{t - 2^\ell k}{2^\ell}\right) \quad \ell, k \in \mathbb{Z},$$

with  $\psi(t)$  as in (12.59), is an orthonormal basis for  $\mathcal{L}^2(\mathbb{R})$ .

*Proof.* Scaling (12.88) using (12.86d), we get that  $V^{(\ell)} = V^{(\ell+1)} \oplus W^{(\ell+1)}$ . Iterating it  $n$  times leads to

$$V^{(\ell)} = W^{(\ell+1)} \oplus W^{(\ell+2)} \oplus \dots \oplus W^{(\ell+n)} \oplus V^{(\ell+n)}.$$

As  $n \rightarrow \infty$  and because of (12.86c), we get<sup>159</sup>

$$V^{(\ell)} = \bigoplus_{i=\ell+1}^{\infty} W^{(i)},$$

and finally, letting  $\ell \rightarrow -\infty$  and because of (12.86b), we obtain

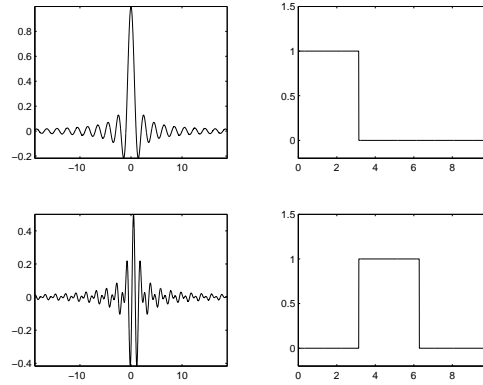
$$\mathcal{L}^2(\mathbb{R}) = \bigoplus_{\ell \in \mathbb{Z}} W^{(\ell)}. \quad (12.90)$$

Since  $\{\psi(t - k)\}_{k \in \mathbb{Z}}$  is an orthonormal basis for  $W^{(0)}$ , by scaling,  $\{\psi_{\ell,k}(t)\}_{k \in \mathbb{Z}}$  is an orthonormal basis for  $W^{(\ell)}$ . Then, following (12.90), the family  $\{\psi_{\ell,k}(t)\}_{\ell,n \in \mathbb{Z}}$  is an orthonormal basis for  $\mathcal{L}^2(\mathbb{R})$ .

Thus, in a fashion complementary to Section 12.1, we obtain a split of  $\mathcal{L}^2(\mathbb{R})$  into a collection  $\{W^{(\ell)}\}_{\ell \in \mathbb{Z}}$  as a consequence of the axioms of multiresolution analysis (12.86a)–(12.86f) (see Figure 12.10 for a graphical representation of the spaces  $V^{(\ell)}$  and  $W^{(\ell)}$ ). We illustrate our discussion with examples.

### Examples

<sup>159</sup>In the infinite sum, we imply closure.



**Figure 12.30:** Sinc scaling function and wavelet. (a) Scaling function  $\varphi(t)$ . (b) Magnitude Fourier transform  $|\Phi(\omega)|$ . (c) Wavelet  $\psi(t)$ . (d) Magnitude Fourier transform  $|\Psi(\omega)|$ .

**EXAMPLE 12.5 (SINC MULTIREOLUTION ANALYSIS)** Let  $V^{(0)}$  be the space of  $\mathcal{L}^2$  functions bandlimited to  $[-\pi, \pi)$ , for which we know that

$$\varphi(t) = \frac{\sin(\pi t)}{\pi t} \quad (12.91)$$

and its integer shifts form an orthonormal basis. Define  $V^{(\ell)}$  to be the space of  $\mathcal{L}^2$  functions bandlimited to  $[-2^{-\ell}\pi, 2^{-\ell}\pi)$ . These are nested spaces of bandlimited functions, which obviously satisfy (12.86a), as they do the axioms of multiresolution analysis (12.86b)–(12.86f), that is, the union of the  $V^{(\ell)}$ s is  $\mathcal{L}^2(\mathbb{R})$ , their intersection is empty, the spaces are scaled versions of each other and are shift invariant with respect to shifts by integer multiples of  $2^\ell$ . The existence of the basis we stated in (12.91). The details are left as Exercise 12.8, including the derivation of the wavelet and the detail spaces  $W^{(\ell)}$ , the spaces of  $\mathcal{L}^2$  bandpass functions,

$$W^{(\ell)} = [-2^{-\ell+1}\pi, -2^{-\ell}\pi) \cup [2^{-\ell}\pi, 2^{-\ell+1}\pi). \quad (12.92)$$

Figure 12.30 shows the sinc scaling function and wavelet both in time as well as Fourier domains.

While the perfect bandpass spaces lead to a bona fide multiresolution analysis of  $\mathcal{L}^2(\mathbb{R})$ , the basis functions have slow decay in time. Since the Fourier transform is discontinuous, the tails of the scaling function and the wavelet decay only as  $O(1/t)$  (as can be seen in the sinc function (12.91)). We will see in latter examples possible remedies to this problem.

**EXAMPLE 12.6 (PIECEWISE-LINEAR MULTIREOLUTION ANALYSIS)** Let  $V^{(0)}$  be the space of continuous  $\mathcal{L}^2$  functions piecewise linear over intervals  $[k, k+1)$ , or  $x(t) \in V^{(0)}$  if  $\|x\| < \infty$  and  $x'(t)$  is piecewise constant over intervals  $[k, k+1)$ . For simplicity, consider functions  $x(t)$  such that  $x(t) = 0$  for  $t < 0$ . Then  $x'(t)$



is specified by the sequence  $\{a_k\}$ , the slopes of  $x(t)$  over intervals  $[k, k+1)$ , for  $k \in \mathbb{N}$ . The nodes of  $x(t)$ , that is, the values at the integers, are given by

$$x(k) = \begin{cases} 0, & k \leq 0; \\ \sum_{i=0}^{k-1} a_i, & k > 0, \end{cases}$$

and the piecewise-linear function is

$$x(t) = [x(k+1) - x(k)](t - k) + x(k) = a_k(t - k) + \sum_{i=0}^{k-1} a_i \quad (12.93)$$

for  $t \in [k, k+1)$  (see Figure 12.31).

The spaces  $V^{(\ell)}$  are simply scaled versions of  $V^{(0)}$ ; they contain functions that are continuous and piecewise linear over intervals  $[2^\ell k, 2^\ell(k+1))$ . Let us verify the axioms of multiresolution.

- (i) *Embedding*: Embedding as in (12.86a) is clear.
- (ii) *Upward Completeness*: Similarly to the piecewise-constant case, piecewise-linear functions are dense in  $\mathcal{L}^2(\mathbb{R})$  (see Footnote 157), and thus upward completeness (12.86b) holds.
- (iii) *Downward Completeness*: Conversely, as  $\ell \rightarrow \infty$ , the approximation gets coarser and coarser, ultimately verifying downward completeness (12.86c).
- (iv) *Scale Invariance*: Scaling (12.86d) is clear from the definition of the piecewise-linear functions over intervals scaled by powers of 2.
- (v) *Shift Invariance*: Similarly, shift invariance (12.86e) is clear from the definition of the piecewise linear functions over intervals scaled by powers of 2.
- (vi) *Existence of a Basis*: What remains is to find a basis for  $V^{(0)}$ . As an educated guess, take the hat function from (3.49a) shifted by 1 to the right and call it  $\theta(t)$  (see Figure 12.32(a)). Then  $x(t)$  in (12.93) can be written as

$$x(t) = \sum_{k=0}^{\infty} b_k \theta(t - k),$$

with  $b_0 = a_0$  and  $b_k = a_k + b_{k-1}$ . We prove this as follows: First,

$$\theta'(t) = \varphi_h(t) - \varphi_h(t-1),$$

where  $\varphi_h(t)$  is the Haar scaling function, the indicator function of the unit interval. Thus,  $x'(t)$  is piecewise constant. Then, the value of the constant between  $k$  and  $k+1$  is  $(b_k - b_{k-1})$  and thus equals  $a_k$  as desired. The only detail is that  $\theta(t)$  is clearly not orthogonal to its integer translates, since

$$\langle \theta(t), \theta(t-k) \rangle_t = \begin{cases} 2/3, & k=0; \\ 1/6, & k=-1, 1; \\ 0, & \text{otherwise.} \end{cases}$$

We can apply the orthogonalization procedure in (E12.3-1). The  $z$ -transform of the sequence  $[1/6 \quad 2/3 \quad 1/6]$  is

$$\frac{1}{6}(z + 4 + z^{-1}) \stackrel{(a)}{=} \frac{2 + \sqrt{3}}{6} \underbrace{[1 + (2 - \sqrt{3})z]}_{\text{right sided}} \underbrace{[1 + (2 - \sqrt{3})z^{-1}]}_{\text{left sided}},$$

where (a) follows from it being a deterministic autocorrelation, positive on the unit circle, and could thus be factored into its spectral roots. Choosing just the right-sided part, with the impulse response

$$\alpha_k = \sqrt{\frac{2 + \sqrt{3}}{6}} (-1)^k (2 - \sqrt{3})^k,$$

leads to

$$\varphi_c(t) = \sum_{k=0}^{\infty} \alpha_k \theta(t - k),$$

a function such that  $\varphi_c(t) = 0$  for  $t < 0$  and orthonormal to its integer translates. It is piecewise linear over integer pieces, but of infinite extent (see Figure 12.32).

Instead of the spectral factorization, we can just take the square root as in (E12.3-1). In Fourier domain,

$$\frac{2}{3} + \frac{1}{6} e^{j\omega} + \frac{1}{6} e^{-j\omega} = \frac{1}{3} (2 + \cos(\omega)).$$

Then,

$$\Phi_s(\omega) = \frac{\sqrt{3} \theta(\omega)}{(2 + \cos(\omega))^{1/2}}$$

is the Fourier transform of a symmetric and orthogonal scaling function  $\varphi_s(t)$  (see Figure 12.32(c)).

Because of the embedding of the spaces  $V^{(\ell)}$ , the scaling functions all satisfy two-scale equations (Exercise 12.6). Once the two-scale equation coefficients are derived, the wavelet can be calculated in the standard manner. Naturally, since the wavelet is a basis for the orthogonal complement of  $V^{(0)}$  in  $V^{(-1)}$ , it will be piecewise linear over half-integer intervals (Exercise 12.7).

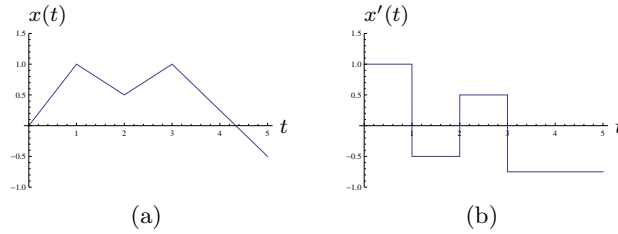
**EXAMPLE 12.7 (MEYER MULTIREOLUTION ANALYSIS)** The idea behind Meyer's wavelet construction is to smooth the sinc solution in Fourier domain, so as to obtain faster decay of the basis functions in the time domain. The simplest way to do this is to allow the Fourier transform magnitude of the scaling function,  $|\Phi(\omega)|^2$ , to linearly decay to zero, that is,

$$|\Phi(\omega)|^2 = \begin{cases} 1, & |\omega| < \frac{2\pi}{3}; \\ 2 - \frac{3|\omega|}{2\pi}, & \frac{2\pi}{3} < |\omega| < \frac{4\pi}{3}. \end{cases} \quad (12.94)$$

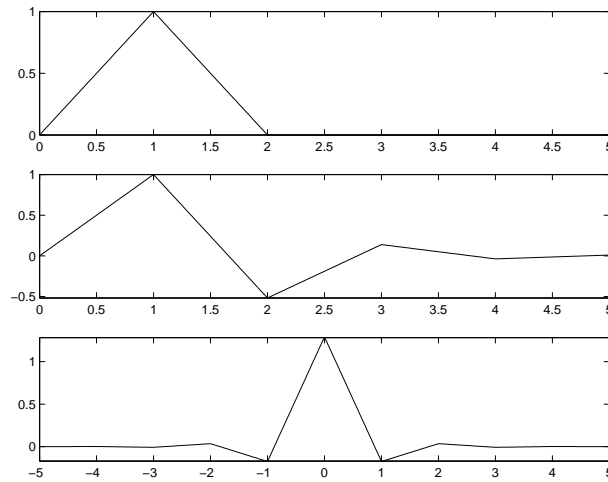
We start by defining a function orthonormal to its integer translates and the space  $V^{(0)}$  spanned by those, axiom (vi).

## 12.3. Wavelet Series

835



**Figure 12.31:** (a) A continuous and piecewise-linear function  $x(t)$  and (b) its derivative  $x'(t)$ .

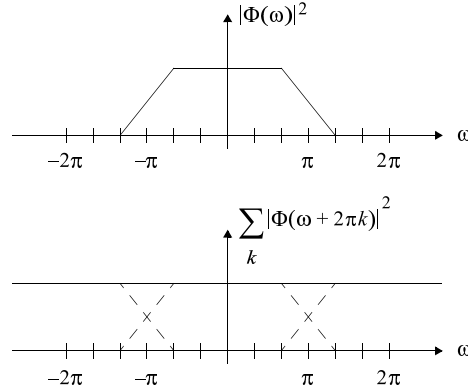


**Figure 12.32:** Basis function for piecewise-linear spaces. (a) The nonorthogonal basis function  $\theta(t)$ . (b) An orthogonalized basis function  $\varphi_c(t)$  such that  $\varphi_c(t) = 0$  for  $t < 0$ . (c) An orthogonalized symmetric basis function  $\varphi_s(t)$ .

- (i) *Existence of a Basis:* The basis function is shown in Figure 12.33, where we also show graphically that (3.73d) holds, proving that  $\{\varphi(t - k)\}_{k \in \mathbb{Z}}$  is an orthonormal set. We now define  $V^{(0)}$  to be

$$V^{(0)} = \text{span}(\{\varphi(t - k)\}_{k \in \mathbb{Z}}).$$

- (ii) *Upward Completeness:* Define  $V^{(\ell)}$  as the scaled version of  $V^{(0)}$ . Then (12.86b) holds, similarly to the sinc case.
- (iii) *Downward Completeness:* Again, (12.86c) holds.
- (iv) *Scale Invariance:* Holds by construction.
- (v) *Shift Invariance:* Holds by construction.
- (vi) *Embedding:* To check  $V^{(0)} \subset V^{(-1)}$  we use Figure 12.34 to see that  $V^{(0)}$  is perfectly represented in  $V^{(-1)}$ . This means we can find a  $2\pi$ -periodic



**Figure 12.33:** Meyer scaling function, with a piecewise linear squared Fourier transform magnitude. (a) The function  $|\Phi(\omega)|^2$ . (b) Proof of orthogonality by verifying (3.73d).

function  $G(e^{j\omega})$  to satisfy the two-scale equation in Fourier domain (12.47), illustrated in Figure 12.35.

Now that we have verified the axioms of multiresolution analysis, we can construct the wavelet. From (12.94), (12.47) and the figure, the DTFT of the discrete-time filter  $g_n$  is

$$|G(e^{j\omega})| = \begin{cases} \sqrt{2}, & |\omega| \leq \frac{\pi}{3}; \\ \sqrt{4 - \frac{6|\omega|}{\pi}}, & \frac{\pi}{3} \leq |\omega| < \frac{2\pi}{3}; \\ 0, & \frac{2\pi}{3} < |\omega| \leq \pi. \end{cases} \quad (12.95)$$

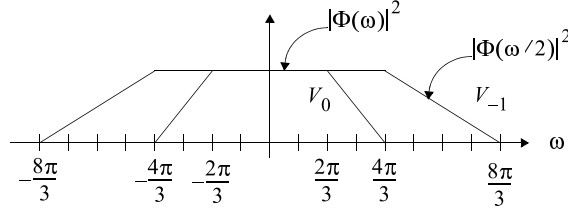
As the phase is not specified, we chose it to be zero making  $G(e^{j\omega})$  real and symmetric. Such a filter has an infinite impulse response, and its  $z$ -transform is not rational (since it is exactly zero over an interval of nonzero measure). It does satisfy, however, the quadrature formula for an orthogonal lowpass filter from (7.13). Choosing the highpass filter in the standard way, (7.24),

$$H(e^{j\omega}) = e^{-j\omega} G(e^{j(\omega+\pi)}),$$

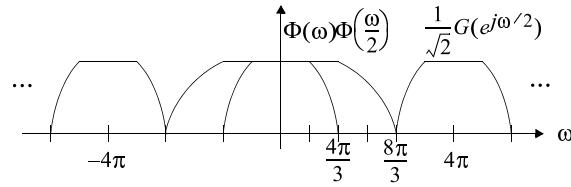
with  $G(e^{j\omega})$  real, and using the two-scale equation for the wavelet in Fourier domain, (12.58), we get

$$\Psi(\omega) = \begin{cases} 0, & |\omega| < \frac{2\pi}{3}; \\ e^{-j\omega} \sqrt{\frac{3\omega}{2\pi} - 1}, & \frac{2\pi}{3} \leq |\omega| < \frac{4\pi}{3}; \\ e^{-j\omega} \sqrt{2 - \frac{3\omega}{4\pi}}, & \frac{4\pi}{3} \leq |\omega| < \frac{8\pi}{3}; \\ 0, & |\omega| \geq \frac{8\pi}{3}. \end{cases} \quad (12.96)$$

The construction and resulting wavelet (a bandpass function) are shown in Figure 12.36. Finally, the scaling function  $\varphi(t)$  and wavelet  $\psi(t)$  are shown, together with their Fourier transforms, in Figure 12.37.



**Figure 12.34:** Embedding  $V^{(0)} \subset V^{(-1)}$  for the Meyer wavelet.



**Figure 12.35:** The two-scale equation for the Meyer wavelet in frequency domain. Note how the  $4\pi$ -periodic function  $G(e^{j(\omega/2+\pi)})$  carves out  $\Phi(\omega)$  from  $\Phi(\omega/2)$ .

The example above showed all the ingredients of the general construction of Meyer wavelets. The key was the orthogonality relation for  $\Phi(\omega)$ , the fact that  $\Phi(\omega)$  is continuous, and that the spaces  $V^{(\ell)}$  are embedded. Since  $\Phi(\omega)$  is continuous,  $\varphi(t)$  decays as  $O(1/t^2)$ . Smoother  $\Phi(\omega)$ 's can be constructed, leading to faster decay of  $\varphi(t)$  (Exercise 12.9).

### 12.3.4 Biorthogonal Wavelet Series

Instead of one scaling function and one wavelet, we now seek two scaling functions,  $\varphi(t)$  and  $\tilde{\varphi}(t)$ , as well as two corresponding wavelets,  $\psi(t)$  and  $\tilde{\psi}(t)$  as in Section 12.2.4, such that the families

$$\psi_{\ell,k}(t) = \frac{1}{\sqrt{2^\ell}} \psi\left(\frac{t - 2^\ell k}{2^\ell}\right), \quad (12.97a)$$

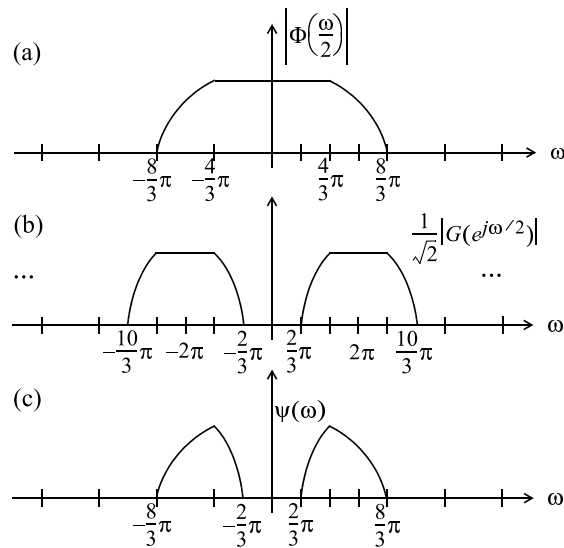
$$\tilde{\psi}_{\ell,k}(t) = \frac{1}{\sqrt{2^\ell}} \tilde{\psi}\left(\frac{t - 2^\ell k}{2^\ell}\right), \quad (12.97b)$$

for  $\ell, k \in \mathbb{Z}$ , form a biorthogonal set

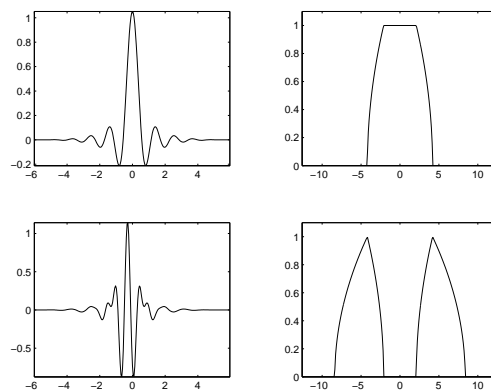
$$\langle \psi_{k,\ell}(t), \tilde{\psi}_{m,n}(t) \rangle = \delta_{n-\ell} \delta_{m-k},$$

and are complete in  $\mathcal{L}^2(\mathbb{R})$ . That is, any  $x(t) \in \mathcal{L}^2(\mathbb{R})$  can be written as either

$$x(t) = \sum_{\ell \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} \beta_k^{(\ell)} \psi_{\ell,k}(t), \quad \beta_k^{(\ell)} = \langle x, \tilde{\psi}_{\ell,k} \rangle,$$



**Figure 12.36:** Construction of the wavelet from the two-scale equation. (a) The stretched scaling function  $\Phi(\omega/2)$ . (b) The stretched and shifted lowpass filter  $G(e^{j(\omega/2+\pi)})$ . (c) The resulting bandpass wavelet  $\Psi(\omega)$ .



**Figure 12.37:** Meyer scaling function and wavelet. (a)  $\varphi(t)$ . (b)  $\Phi(\omega)$ . (c)  $\psi(t)$ . (d)  $\Psi(\omega)$ .

or

$$x(t) = \sum_{\ell \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} \tilde{\beta}_k^{(\ell)} \tilde{\psi}_{\ell,k}(t), \quad \tilde{\beta}_k^{(\ell)} = \langle x, \psi_{\ell,k} \rangle.$$

## 12.3. Wavelet Series

839

These scaling functions and wavelets will satisfy two-scale equations as before

$$\begin{aligned}\varphi(t) &= \sqrt{2} \sum_{n \in \mathbb{Z}} g_n \varphi(2t - n), & \tilde{\varphi}(t) &= \sqrt{2} \sum_{n \in \mathbb{Z}} \tilde{g}_n \tilde{\varphi}(2t - n), \\ \psi(t) &= \sqrt{2} \sum_{n \in \mathbb{Z}} h_n \varphi(2t - n), & \tilde{\psi}(t) &= \sqrt{2} \sum_{n \in \mathbb{Z}} \tilde{h}_n \tilde{\varphi}(2t - n).\end{aligned}$$

We can then define a biorthogonal multiresolution analysis by

$$V^{(0)} = \text{span}(\{\varphi(t - k)\}_{k \in \mathbb{Z}}), \quad \tilde{V}^{(0)} = \text{span}(\{\tilde{\varphi}(t - k)\}_{k \in \mathbb{Z}}),$$

and the appropriate scaled spaces

$$V^{(\ell)} = \text{span}(\{\varphi_{\ell,k}\}_{k \in \mathbb{Z}}), \quad \tilde{V}^{(\ell)} = \text{span}(\{\tilde{\varphi}_{\ell,k}\}_{k \in \mathbb{Z}}), \quad (12.98)$$

for  $\ell \in \mathbb{Z}$ . For a given  $\varphi(t)$ —for example, the hat function—we can verify that the axioms of multiresolution analysis (Exercise 12.14). From there, define the wavelet families as in (12.97a)–(12.97b), which then lead to the wavelet spaces  $W^{(\ell)}$  and  $\tilde{W}^{(\ell)}$ . While this seems very natural, the geometry is more complicated than in the orthogonal case. On the one hand, we have the decompositions

$$V^{(\ell)} = V^{(\ell+1)} \oplus W^{(\ell+1)}, \quad (12.99)$$

$$\tilde{V}^{(\ell)} = \tilde{V}^{(\ell+1)} \oplus \tilde{W}^{(\ell+1)}, \quad (12.100)$$

as can be verified by using the two-scale equations for the scaling functions and wavelets involved. On the other hand, unlike the orthonormal case,  $V^{(\ell)}$  is not orthogonal to  $W^{(\ell)}$ . Instead,

$$\tilde{W}^{(\ell)} \perp V^{(\ell)}, \quad W^{(\ell)} \perp \tilde{V}^{(\ell)},$$

similarly to a biorthogonal basis (see Figure 7.11). We explore these relationships in Exercise 12.15 to show that

$$\tilde{W}^{(\ell)} \perp W^{(m)}, \quad \ell \neq m.$$

The embedding (12.86a) has then two forms:

$$\dots \subset V^{(2)} \subset V^{(1)} \subset V^{(0)} \subset V^{(-1)} \subset V^{(-2)} \subset \dots,$$

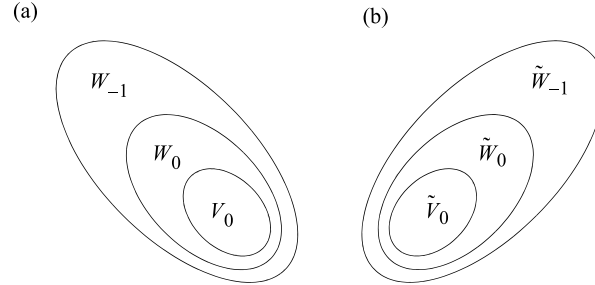
with detail spaces  $\{W^{(\ell)}\}_{\ell \in \mathbb{Z}}$ , or,

$$\dots \subset \tilde{V}^{(2)} \subset \tilde{V}^{(1)} \subset \tilde{V}^{(0)} \subset \tilde{V}^{(-1)} \subset \tilde{V}^{(-2)} \subset \dots,$$

with detail spaces  $\{\tilde{W}^{(\ell)}\}_{\ell \in \mathbb{Z}}$ . The detail spaces allow us to write

$$\mathcal{L}^2(\mathbb{R}) = \bigoplus_{\ell \in \mathbb{Z}} W^{(\ell)} = \bigoplus_{\ell \in \mathbb{Z}} \tilde{W}^{(\ell)}.$$

The diagram in Figure 12.38 illustrates these two splits and the biorthogonality between them.



**Figure 12.38:** The space  $\mathcal{L}^2(\mathbb{R})$  is split according to two different embeddings. (a) Embedding  $V^{(\ell)}$  based on the scaling function  $\varphi(t)$ . (b) Embedding  $\tilde{V}^{(\ell)}$  based on the dual scaling function  $\tilde{\varphi}(t)$ . Note that orthogonality is “across” the spaces and their duals, for example,  $\tilde{W}^{(\ell)} \perp V^{(\ell)}$ .

## 12.4 Wavelet Frame Series

### 12.4.1 Definition of the Wavelet Frame Series

### 12.4.2 Frames from Sampled Wavelet Series

## 12.5 Continuous Wavelet Transform

### 12.5.1 Definition of the Continuous Wavelet Transform

The continuous wavelet transform uses a function  $\psi(t)$  and all its shifted and scaled versions to analyze functions. Here we consider only real wavelets; this can be extended to complex wavelets without too much difficulty.

Consider a real wavelet  $\psi(t) \in \mathcal{L}^2(\mathbb{R})$  centered around  $t = 0$  and having at least one zero moment (i.e.,  $\int \psi(t) dt = 0$ ). Now, consider all its shifts and scales, denoted by

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right), \quad a \in \mathbb{R}^+, \quad b \in \mathbb{R}, \quad (12.101)$$

which means that  $\psi_{a,b}(t)$  is centered around  $b$  and scaled by a factor  $a$ . The scale factor  $\frac{1}{\sqrt{a}}$  insures that the  $\mathcal{L}^2$  norm is preserved, and without loss of generality, we can assume  $\|\psi\| = 1$  and thus

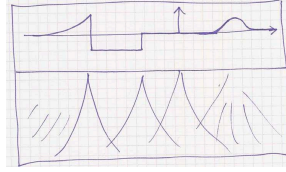
$$\|\psi_{a,b}\| = 1.$$

There is one more condition on the wavelet, namely the *admissibility condition* stating that the Fourier transform  $\Psi(\omega)$  must satisfy

$$C_\psi = \int_{\omega \in \mathbb{R}^+} \frac{|\Psi(\omega)|^2}{|\omega|} d\omega < \infty. \quad (12.102)$$

Since  $|\Psi(0)| = 0$  because of the zero moment property, this means that  $|\Psi(\omega)|$  has to decay for large  $\omega$ , which it will if  $\psi$  has any smoothness. In short, (12.102) is a





**Figure 12.39:** The wavelet transform. (a) An example function. (b) The magnitude of wavelet transform  $|X(a, b)|$ .

very mild requirement that is satisfied by all wavelets of interest (see, for example, Exercise 12.10). Now, given a function  $x(t)$  in  $\mathcal{L}^2(\mathbb{R})$ , we can define its continuous wavelet transform as

$$\begin{aligned} X(a, b) &= \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} \psi\left(\frac{t-b}{a}\right) x(t) dt = \int_{-\infty}^{\infty} \psi_{a,b}(t) x(t) dt \\ &= \langle f, \psi_{a,b} \rangle. \end{aligned} \quad (12.103)$$

In words, we take the inner product of the function  $x(t)$  with a wavelet centered at location  $b$ , and rescaled by a factor  $a$ , shown in Figure 12.16. A numerical example is given in Figure 12.39, which displays the magnitude  $|X(a, b)|$  as an image. It is already clear that the continuous wavelet transform acts as a singularity detector or derivative operator, and that smooth regions are suppressed, which follows from the zero moment property.

Let us rewrite the continuous wavelet transform at scale  $a$  as a convolution. For this, it will be convenient to introduce the scaled and normalized version of the wavelet,

$$\psi_a(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t}{a}\right) \quad \xleftrightarrow{\text{FT}} \quad \Psi_a(\omega) = \sqrt{a} \Psi(a\omega), \quad (12.104)$$

as well as the notation  $\bar{\psi}(t) = \psi(-t)$ . Then

$$\begin{aligned} X(a, b) &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) f(t) dt = \int_{-\infty}^{\infty} \psi_a(t-b) f(t) dt \\ &= (f * \bar{\psi}_a)(b). \end{aligned} \quad (12.105)$$

Now the Fourier transform of  $X(a, b)$  over the “time” variable  $b$  is

$$X(a, \omega) = X(\omega) \Psi_a^*(\omega) = X(\omega) \sqrt{a} \Psi^*(a\omega), \quad (12.106)$$

where we used  $\psi(-t) \xleftrightarrow{\text{FT}} \Psi^*(\omega)$  since  $\psi(t)$  is real.

### 12.5.2 Existence and Convergence of the Continuous Wavelet Transform

The invertibility of the continuous wavelet transform is of course a key result: not only can we compute the continuous wavelet transform, but we are actually able to

come back! This inversion formula was first proposed by J. Morlet.<sup>160</sup>

**PROPOSITION 12.13 (INVERSION OF THE CONTINUOUS WAVELET TRANSFORM)**  
Consider a real wavelet  $\psi$  satisfying the admissibility condition (12.102). A function  $f \in \mathcal{L}^2(\mathbb{R})$  can be recovered from its continuous wavelet transform  $X(a, b)$  by the inversion formula

$$x(t) = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty X(a, b) \psi_{a,b}(t) \frac{db da}{a^2}, \quad (12.107)$$

where equality is in the  $\mathcal{L}^2$  sense.

*Proof.* Denote the right hand side of (12.107) by  $x(t)$ . In that expression, we replace  $X(a, b)$  by (12.105) and  $\psi_{a,b}(t)$  by  $\psi_a(t - b)$  to obtain

$$\begin{aligned} x(t) &= \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty (f * \bar{\psi}_a)(b) \psi_a(t - b) \frac{db da}{a^2} \\ &= \frac{1}{C_\psi} \int_0^\infty (f * \bar{\psi}_a * \psi_a)(t) \frac{da}{a^2}, \end{aligned}$$

where the integral over  $b$  was recognized as a convolution. We will show the  $\mathcal{L}^2$  equality of  $x(t)$  and  $x(t)$  through the equality of their Fourier transforms. The Fourier transform of  $x(t)$  is

$$\begin{aligned} X(\omega) &= \frac{1}{C_\psi} \int_{-\infty}^\infty \int_0^\infty (f * \bar{\psi}_a * \psi_a)(t) e^{-j\omega t} \frac{da dt}{a^2} \\ &\stackrel{(a)}{=} \frac{1}{C_\psi} \int_0^\infty X(\omega) \Psi_a^*(\omega) \Psi_a(\omega) \frac{da}{a^2} \\ &\stackrel{(b)}{=} \frac{1}{C_\psi} X(\omega) \int_0^\infty a |\Psi(a\omega)|^2 \frac{da}{a^2}, \end{aligned} \quad (12.108)$$

where (a) we integrated first over  $t$ , and transformed the two convolutions into products; and (b) we used (12.104). In the remaining integral above, apply a change of variable  $\Omega = a\omega$  to compute:

$$\int_0^\infty |\Psi(a\omega)|^2 \frac{da}{a} = \int_0^\infty \frac{|\Psi(\Omega)|^2}{\Omega} d\Omega = C_\psi, \quad (12.109)$$

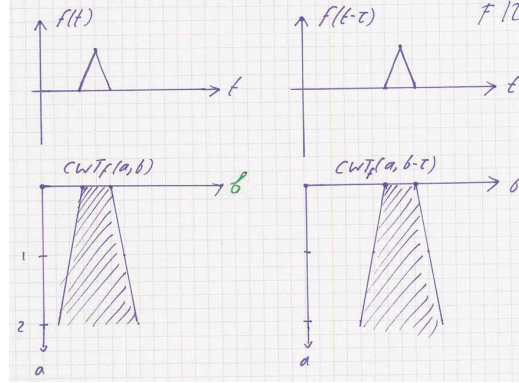
which together with (12.108), shows that  $X(\omega) = X(\omega)$ . By Fourier inversion, we have proven that  $x(t) = x(t)$  in the  $\mathcal{L}^2$  sense.

The formula (12.107) is also sometimes called the *resolution of the identity* and goes back to Calderon in the 1960's in a context other than wavelets.

### 12.5.3 Properties of the Continuous Wavelet Transform

#### Linearity

<sup>160</sup>The story goes that Morlet asked a mathematician for a proof, but only got as an answer: "This formula, being so simple, would be known if it were correct."



**Figure 12.40:** The shift property of the continuous wavelet transform.

**Shift in Time** The continuous wavelet transform has a number of properties, several of these being extensions or generalizations of properties seen already for wavelet series. Let us start with shift and scale invariance. Consider  $g(t) = x(t - \tau)$ , or a delayed version of  $x(t)$ . Then

$$\begin{aligned} X_g(a, b) &= \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} \psi\left(\frac{t-b}{a}\right) x(t - \tau) dt = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} \psi\left(\frac{t' + \tau - b}{a}\right) x(t') dt' \\ &= X_f(a, b - \tau) \end{aligned} \quad (12.110)$$

by using the change of variables  $t' = t - \tau$ . That is, the continuous wavelet transform of  $g$  is simply a delayed version of the wavelet transform of  $x(t)$ , as shown in Figure 12.40.

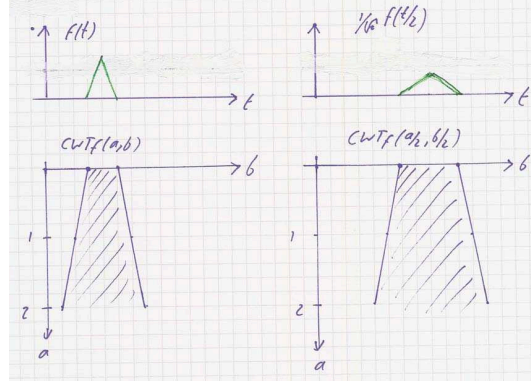
**Scaling in Time** For the scaling property, consider a scaled and normalized version of  $x(t)$ ,

$$g(t) = \frac{1}{\sqrt{s}} f\left(\frac{t}{s}\right),$$

where the renormalization ensures that  $\|g\| = \|f\|$ . Computing the continuous wavelet transform of  $g$ , using the change of variables  $t' = t/s$ , gives

$$\begin{aligned} X_g(a, b) &= \frac{1}{\sqrt{as}} \int_{-\infty}^{\infty} \psi\left(\frac{t-b}{a}\right) f\left(\frac{t}{s}\right) dt = \frac{1}{\sqrt{as}} \int_{-\infty}^{\infty} \psi\left(\frac{st' - b}{a}\right) f(t') dt' \\ &= \sqrt{\frac{s}{a}} \int_{-\infty}^{\infty} \psi\left(\frac{t' - b/s}{a/s}\right) x(t') dt' = X\left(\frac{a}{s}, \frac{b}{s}\right). \end{aligned} \quad (12.111)$$

In words: if  $g(t)$  is a version of  $x(t)$  scaled by a factor  $s$  and normalized to maintain its energy, then its continuous wavelet transform is scaled by  $s$  both in  $a$  and  $b$ . A graphical representation of the scaling property is shown in Figure 12.41.



**Figure 12.41:** The scaling property of the continuous wavelet transform.

Consider now a function  $x(t)$  with unit energy and having its wavelet transform concentrated mostly in a unit square, say  $[a_0, a_0 + 1] \times [b_0, b_0 + 1]$ . The continuous wavelet transform of  $g(t)$  is then mostly concentrated in a square  $[sa_0, s(a_0 + 1)] \times [sb_0, s(b_0 + 1)]$ , a cell of area  $s^2$ . But remember that  $g(t)$  has still unit energy, while its continuous wavelet transform now covers a surface increased by  $s^2$ . Therefore, when evaluating an energy measure in the continuous wavelet transform domain, we need to renormalize by a factor  $a^2$ , as was seen in both the inversion formula (12.107) and the energy conservation formula (12.112).

When comparing the above properties with the equivalent ones from wavelet series, the major difference is that shift and scale are arbitrary real variables, rather than constrained, dyadic rationals (powers of 2 for the scale, multiples of the scale for shifts). Therefore, we obtain true time scale and shift properties.

**Parseval's Equality** Closely related to the resolution of the identity is an energy conservation formula, an analogue to Parseval's equality.

**PROPOSITION 12.14 (ENERGY CONSERVATION OF THE CONTINUOUS WAVELET TRANSFORM)**

Consider a function  $f \in \mathcal{L}^2(\mathbb{R})$  and its continuous wavelet transform  $X(a, b)$  with respect to a real wavelet  $\psi$  satisfying the admissibility condition (12.102). Then, the following energy conservation holds:

$$\int_{-\infty}^{\infty} |x(t)|^2 dt = \frac{1}{C_\psi} \int_{a \in \mathbb{R}^+} \int_{b \in \mathbb{R}} |X(a, b)|^2 \frac{db da}{a^2}. \quad (12.112)$$

## 12.5. Continuous Wavelet Transform

845

*Proof.* Expand the right hand side (without the leading constant) as

$$\begin{aligned} \int_{a \in \mathbb{R}^+} \int_{b \in \mathbb{R}} |X(a, b)|^2 \frac{db da}{a^2} &\stackrel{(a)}{=} \int_{a \in \mathbb{R}^+} \int_{b \in \mathbb{R}} |(f * \bar{\psi}_a)(b)|^2 db \frac{da}{a^2} \\ &\stackrel{(b)}{=} \int_{a \in \mathbb{R}^+} \frac{1}{2\pi} \int_{b \in \mathbb{R}} |X(\omega) \sqrt{a} \Psi^*(a\omega)|^2 d\omega \frac{da}{a^2} \\ &= \int_{a \in \mathbb{R}^+} \frac{1}{2\pi} \int_{b \in \mathbb{R}} |X(\omega)|^2 |\Psi(a\omega)|^2 d\omega \frac{da}{a}, \end{aligned}$$

where (a) uses (12.105); and (b) uses Parseval's equality for the Fourier transform with respect to  $b$ , also transforming the convolution into a product. Changing the order of integration and in (c) using the change of variables  $\Omega = a\omega$  allows us to write the above as

$$\begin{aligned} \int_{a \in \mathbb{R}^+} \int_{b \in \mathbb{R}} |X(a, b)|^2 \frac{db da}{a^2} &= \int_{-\infty}^{\infty} \frac{1}{2\pi} |X(\omega)|^2 \int_{a \in \mathbb{R}^+} |\Psi(a\omega)|^2 \frac{da}{a^2} d\omega \\ &\stackrel{(c)}{=} \int_{\omega \in \mathbb{R}} \frac{1}{2\pi} |F(\omega)|^2 \underbrace{\int_{\omega \in \mathbb{R}^+} |\Psi(\Omega)|^2 \frac{d\Omega}{\Omega}}_{C_\psi} d\omega. \end{aligned}$$

Therefore

$$\frac{1}{C_\psi} \int_{a \in \mathbb{R}^+} \int_{b \in \mathbb{R}} |X(a, b)|^2 \frac{db da}{a^2} = \frac{1}{2\pi} \int_{\omega \in \mathbb{R}} |X(\omega)|^2 d\omega,$$

and applying Parseval's equality to the right side proves (12.112).

Both the inversion formula and the energy conservation formula use  $da db/a^2$  as an integration measure. This is related to the scaling property of the continuous wavelet transform as will be shown below. Note that the extension to a complex wavelet is not hard; the integral over  $da$  has to go from  $-\infty$  to  $\infty$ , and  $C_\psi$  has to be defined accordingly.

**Redundancy** The continuous wavelet transform maps a one-dimensional function into a two-dimensional one: this is clearly very redundant. In other words, only a small subset of two-dimensional functions correspond to wavelet transforms. We are thus interested in characterizing the image of one-dimensional functions in the continuous wavelet transform domain.

A simple analogue is in order. Consider an  $M$  by  $N$  matrix  $T$  having orthonormal columns (i.e.,  $T^T T = I$ ) with  $M > N$ . Suppose  $y$  is the image of an arbitrary vector  $x \in \mathbb{R}^N$  through the operator  $T$ , or  $y = Tx$ . Clearly  $y$  belongs to a subspace  $S$  of  $\mathbb{R}^M$ , namely the span of the columns of  $T$ .

There is a simple test to check if an arbitrary vector  $z \in \mathbb{R}^M$  belongs to  $S$ . Introduce the kernel matrix  $K$ ,

$$K = TT^T, \quad (12.113)$$

which is the  $M$  by  $M$  matrix of outer products of the columns of  $T$ . Then, a vector  $z$  belong to  $S$  if and only if it satisfies

$$Kz = z. \quad (12.114)$$

Indeed, if  $z$  is in  $S$ , then it can be written as  $z = Tx$  for some  $x$ . Substituting this into the left side of (12.114) leads to

$$Kz = TT^T Tx = Tx = z.$$

Conversely, if (12.114) holds then  $z = Kz = TT^T z = Tx$ , showing that  $z$  belongs to  $S$ .

If  $z$  is not in  $S$ , then  $Kz = \hat{z}$  is the orthogonal projection of  $z$  onto  $S$  as can be verified. See Exercise 12.11 for a discussion of this, as well as the case of non-orthonormal columns in  $T$ .

We now extend the test given in (12.114) to the case of the continuous wavelet transform. For this, let us introduce the *reproducing kernel* of the wavelet  $\psi(t)$ , defined as

$$K(a_0, b_0, a, b) = \langle \psi_{a_0, b_0}, \psi_{a, b} \rangle. \quad (12.115)$$

This is the deterministic crosscorrelation of two wavelets at scale and shifts  $(a_0, b_0)$  and  $(a, b)$ , respectively, and is the equivalent of the matrix  $K$  in (12.113).

Call  $V$  the space of functions  $X(a, b)$  that are square integrable with respect to the measure  $(db da)/a^2$  (see also Proposition 12.14). In this space, there exists a subspace  $S$  that corresponds to bona fide continuous wavelet transforms. Similarly to what we just did in finite dimensions, we give a test to check whether a function  $X(a, b)$  in  $V$  actually belongs to  $S$ , that is, if it is the continuous wavelet transform of some one-dimensional function  $x(t)$ .

**PROPOSITION 12.15 (REPRODUCING KERNEL PROPERTY OF THE CONTINUOUS WAVELET TRANSFORM)**  
A function  $X(a, b)$  is the continuous wavelet transform of a function  $x(t)$  if and only if it satisfies

$$X(a_0, b_0) = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty K(a_0, b_0, a, b) X(a, b) \frac{db da}{a^2}. \quad (12.116)$$

*Proof.* We show that if  $X(a, b)$  is a continuous wavelet transform of some function  $x(t)$ , then (12.116) holds. Completing the proof by showing that the converse is also true is left as Exercise 12.12.

By assumption,

$$X(a_0, b_0) = \int_{-\infty}^\infty \psi_{a_0, b_0}(t) x(t) dt.$$

Replace  $x(t)$  by its inversion formula (12.107), or

$$\begin{aligned} F(a_0, b_0) &= \int_{-\infty}^\infty \psi_{a_0, b_0}(t) \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty \psi_{a, b}(t) X(a, b) \frac{db da}{a^2} dt \\ &\stackrel{(a)}{=} \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty \int_{-\infty}^\infty \psi_{a_0, b_0}(t) \psi_{a, b}(t) X(a, b) dt \frac{db da}{a^2} \\ &\stackrel{(b)}{=} \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty K(a_0, b_0, a, b) X(a, b) \frac{db da}{a^2}, \end{aligned}$$

where (a) we interchanged the order of integration; and (b) we integrated over  $t$  to get the reproducing kernel (12.115).

**Characterization of Singularities** The continuous wavelet transform has an interesting localization property which is related to the fact that as  $a \rightarrow 0$ , the wavelet  $\psi_{a,b}(t)$  becomes arbitrarily narrow, performing a zoom in the vicinity of  $b$ . This is easiest to see for  $x(t) = \delta(t - \tau)$ . Then

$$X(a, b) = \frac{1}{\sqrt{a}} \int_{t \in \mathbb{R}} \psi\left(\frac{t-b}{a}\right) \delta(t - \tau) dt = \frac{1}{\sqrt{a}} \psi\left(\frac{\tau-b}{a}\right). \quad (12.117)$$

This is the wavelet scaled by  $a$  and centered at  $b$ . As  $a \rightarrow 0$ , the continuous wavelet transform narrows exactly on the singularity and grows as  $a^{-1/2}$ .

A similar behavior can be shown for other singularities as well, which we do now. For simplicity, we consider a compactly supported wavelet with  $N$  zero moments. We have seen the most elementary case, namely the Haar wavelet (with a single zero moment) in Section 12.1. Another example is the ramp function starting at  $\tau$ :

$$x(t) = \begin{cases} 0, & t \leq \tau; \\ t - \tau, & t > \tau. \end{cases}$$

This function is continuous, but its derivative is not. Actually, its second derivative is a Dirac delta function at location  $\tau$ .

To analyze this function and its singularity, we need a wavelet with at least 2 zero moments. Given a compactly supported wavelet, its second order primitive will be compactly supported as well. To compute the continuous wavelet transform  $X(a, b)$ , we can apply integration by parts just like in (12.34) to obtain

$$X(a, b) = - \int_{t \in \mathbb{R}} \sqrt{a} \theta\left(\frac{t-b}{a}\right) x'(t) dt,$$

where  $x'(t)$  is now a step function. We apply integration by parts one more time to get

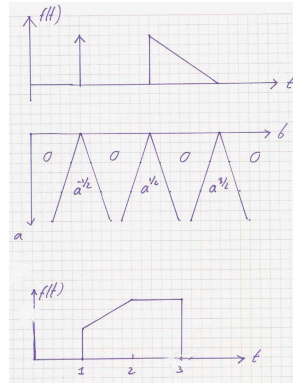
$$\begin{aligned} X(a, b) &= - \left[ a^{3/2} \theta^{(1)}\left(\frac{t-b}{a}\right) x'(t) \right]_{t \in \mathbb{R}} + a^{3/2} \int_{t \in \mathbb{R}} \theta^{(1)}\left(\frac{t-b}{a}\right) x''(t) dt \\ &= a^{3/2} \int_{t \in \mathbb{R}} \theta^{(1)}\left(\frac{t-b}{a}\right) \delta(t - \tau) dt = a^{3/2} \theta^{(1)}\left(\frac{\tau-b}{a}\right), \end{aligned} \quad (12.118)$$

where  $\theta^{(1)}(t)$  is the primitive of  $\theta(t)$ , and the factor  $a^{3/2}$  comes from an additional factor  $a$  due to integration of  $\theta(t/a)$ . The key, of course, is that as  $a \rightarrow 0$ , the continuous wavelet transform zooms towards the singularity and has a behavior of the order  $a^{3/2}$ . These are examples of the following general result.

**PROPOSITION 12.16 (LOCALIZATION PROPERTY OF THE CONTINUOUS WAVELET TRANSFORM)**

Consider a wavelet  $\psi$  of compact support having  $N$  zero moments and a function  $x(t)$  with a singularity of order  $n \leq N$  (meaning the  $n$ th derivative is a Dirac delta function; for example, Dirac delta function = 0, step = 1, ramp = 2, etc.). Then, the wavelet transform in the vicinity of the singularity at  $\tau$  is of the form

$$X(a, b) = (-1)^n a^{n-1/2} \psi^{(n)}\left(\frac{\tau-b}{a}\right), \quad (12.119)$$



**Figure 12.42:** A function with singularities of order 0, 1 and 2, and its wavelet transform.

where  $\psi^{(n)}$  is the  $n$ th primitive of  $\psi$ .

*Proof.* (Sketch) The proof follows the arguments developed above for  $n = 0, 1$ , and  $2$ . Because  $\psi(t)$  has  $N$  zero moments, its primitives of order  $n \leq N$  are also compactly supported. For a singularity of order  $n$ , we apply integration by parts  $n$  times. Each primitive adds a scaling factor  $a$ ; this explains the factor  $a^{n-1/2}$  (the  $-1/2$  comes from the initial  $1/\sqrt{a}$  factor in the wavelet). After  $n$  integrations by parts,  $x(t)$  has been differentiated  $n$  times, is thus a Dirac delta function, and reproduces  $\psi^{(n)}$  at location  $\tau$ .

The key is that the singularities are not only precisely located at small scales, but the behavior of the continuous wavelet transform also indicates the singularity type. Figure 12.42 sketches the continuous wavelet transform of a function with a few singularities.

We considered the behavior around points of singularity, but what about “smooth” regions? Again, assume a wavelet of compact support and having  $N$  zero moments. Clearly, if the function  $x(t)$  is polynomial of order  $N - 1$  or less, all inner products with the wavelet will be exactly zero due to the zero moment property. If the function  $x(t)$  is piecewise polynomial,<sup>161</sup> then the inner product will be zero once the wavelet is inside an interval, while boundaries will be detected according to the types of singularities that appear. We have calculated an example in Section 12.1 for Haar, which makes the above explicit, while also pointing out what happens when the wavelet does not have enough zero moments.

**Decay and Smoothness** Beyond polynomial and piecewise-polynomial functions, let us consider more general smooth functions. Among the many possible classes

<sup>161</sup>That is, the function is a polynomial over intervals  $(t_i, t_{i+1})$ , with singularities at the interval boundaries.



## 12.5. Continuous Wavelet Transform

849

of smooth functions, we consider functions having  $m$  continuous derivatives, or the space  $C^m$ .

For the wavelet, we take a compactly supported wavelet  $\psi$  having  $N$  zero moments. Then, the  $N$ th primitive, denoted  $\psi^{(N)}$ , is compactly supported and

$$\int_{t \in \mathbb{R}} \psi^{(N)}(t) dt = C \neq 0.$$

This follows since the Fourier transform of  $\psi$  has  $N$  zeros at the origin, and each integration removes one, leaving the Fourier transform  $\psi^{(N)}$  nonzero at the origin. For example, the primitive of the Haar wavelet is the hat function in (12.33), with integral equal to  $1/2$ .

Consider the following scaled version of  $\psi^{(N)}$ , namely  $a^{-1}\psi^{(N)}(t/a)$ . This function has an integral equal to  $C$ , and it acts like a Dirac delta function as  $a \rightarrow 0$  in that, for a continuous function  $x(t)$ ,

$$\lim_{a \rightarrow 0} \int_{t \in \mathbb{R}} \frac{1}{a} \psi^{(N)}\left(\frac{t-b}{a}\right) x(t) dt = Cx(b). \quad (12.120)$$

Again, the Haar wavelet with its primitive is a typical example, since a limit of scaled hat functions is a classic way to obtain the Dirac delta function. We are now ready to prove the decay behavior of the continuous wavelet transform as  $a \rightarrow 0$ .

**PROPOSITION 12.17 (DECAY OF CONTINUOUS WAVELET TRANSFORM FOR  $x \in \mathbb{C}^N$ )**

Consider a compactly supported wavelet  $\psi$  with  $N$  zero moments,  $N \geq 1$ , and primitives  $\psi^{(1)}, \dots, \psi^{(N)}$ , where  $\int \psi^{(N)}(t) dt = C$ . Given a function  $x(t)$  having  $N$  continuous and bounded derivatives  $f^{(1)}, \dots, f^{(N)}$ , or  $f \in \mathbb{C}^N$ , then the continuous wavelet transform of  $x(t)$  with respect to  $\psi$  behaves as

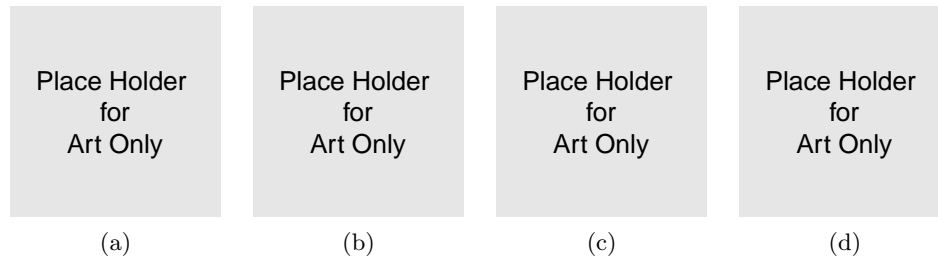
$$|X(a, b)| \leq C' a^{N+1/2} \quad (12.121)$$

for  $a \rightarrow 0$ .

*Proof.* (sketch) The proof closely follows the method of integration by parts as used in Proposition 12.16. That is, we take the  $N$ th derivative of  $x(t)$ ,  $f^{(N)}(t)$ , which is continuous and bounded by assumption. We also have the  $N$ th primitive of the wavelet,  $\psi^{(N)}(t)$ , which is of compact support and has a finite integral. After  $N$  integrations by parts, we have

$$\begin{aligned} X(a, b) &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) x(t) dt \\ &\stackrel{(a)}{=} (-1)^N a^N \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} \psi^{(N)}\left(\frac{t-b}{a}\right) f^{(N)}(t) dt \\ &\stackrel{(b)}{=} (-1)^N a^{N+1/2} \int_{-\infty}^{\infty} \frac{1}{a} \psi^{(N)}\left(\frac{t-b}{a}\right) f^{(N)}(t) dt, \end{aligned}$$

where (a)  $N$  steps of integration by parts contribute a factor  $a^N$ ; and (b) we normalize the  $N$ th primitive by  $1/a$  so that it has a constant integral and acts as a Dirac delta



**Figure 12.43:** A function and its scalogram. (a) Function with various modes. (b) Scalogram with a Daubechies wavelet. (c) Scalogram with a symmetric wavelet. (d) Scalogram with a Morlet wavelet.

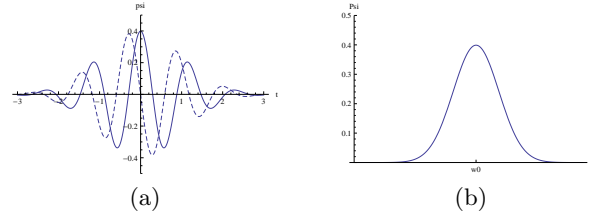
function as  $a \rightarrow 0$ . Therefore, for small  $a$ , the integral above tends towards  $Cf^{(N)}(b)$ , which is finite, and the decay of the continuous wavelet transform is thus of order  $a^{N+1/2}$ .

While we used a global smoothness, it is clear that it is sufficient for  $x(t)$  to be  $C^N$  in the vicinity of  $b$  for the decay to hold. The converse result, namely the necessary decay of the wavelet transform for  $x(t)$  to be in  $C^N$ , is a technical result which is more difficult to prove; it requires non-integer, Lipschitz, regularity. Note that if  $x(t)$  is smoother, that is, it has more than  $N$  continuous derivatives, the decay will still be of order  $a^{N+1/2}$  since we cannot apply more integration by parts steps. Also, the above result is valid for  $N \geq 1$  and thus cannot be applied to functions in  $C^0$ , but it can still be shown that the behavior is of order  $a^{1/2}$  as is to be expected.

**Scalograms** So far, we have only sketched continuous wavelet transforms, to point out general behavior like localization and other relevant properties. For “real” functions, a usual way of displaying the continuous wavelet transform is the density plot of the continuous wavelet transform magnitude  $|X(a, b)|$ . This is done in Figure 12.43 for a particular function and for 3 different wavelets, namely an orthogonal Daubechies wavelet, a symmetric biorthogonal wavelet, and the Morlet wavelet.

As can be seen, the scalograms with respect to symmetric wavelets (Figure 12.43 (c) and (d)) have no drift across scales, which helps identify singularities. The zooming property at small scales is quite evident from the scalogram.

**Remarks** The continuous-time continuous wavelet transform can be seen as a *mathematical microscope*. Indeed, it can zoom in, and describe the local behavior of a function very precisely. This pointwise characterization is a distinguishing feature of the continuous wavelet transform. The characterization itself is related to the wavelet being a local derivative operator. Indeed, a wavelet with  $N$  zero moments acts like an  $N$ th order derivative on the function analyzed by the wavelet transform, as was seen in the proofs of Propositions 12.16 and 12.17. Together with the fact that all scales are considered, this shows that the continuous wavelet transform is a multiscale differential operator.



**Figure 12.44:** Morlet wavelet. (a) Time domain function, with real and imaginary parts in solid and dotted lines, respectively. (b) Magnitude spectrum of the Fourier transform.

*Compactly Supported Wavelets:* Throughout the discussion so far, we have often used the Haar wavelet (actually, its centered version) as the exemplary wavelet used in a continuous wavelet transform. The good news is that it is simple, short, and antisymmetric around the origin. The limitation is that in the frequency domain it has only a single zero at the origin; thus it can only characterize singularities up to order 1, and the decay of the continuous wavelet transform for smooth functions is limited.

Therefore, one can use higher order wavelets, like any of the Daubechies wavelets, or any biorthogonal wavelet. The key is the number of zeros at the origin. The attraction of biorthogonal wavelets is that there are symmetric or antisymmetric solutions. Thus, singularities are well localized along vertical lines, which is not the case for non-symmetric wavelets like the Daubechies wavelets. At the same time, there is no reason to use orthogonal or biorthogonal wavelets, since any functions satisfying the admissibility conditions (12.102) and having a sufficient number of zero moments will do. In the next subsection, scalograms will highlight differences between continuous wavelet transforms using different wavelets.

*Morlet Wavelet:* The classic, and historically first wavelet is a windowed complex exponential, first proposed by Jean Morlet. As a window, a Gaussian bell shape is used, and the complex exponential makes it a bandpass filter. Specifically, the wavelet is given by

$$\psi(t) = \frac{1}{\sqrt{2\pi}} e^{-j\omega_0 t} e^{-t^2/2}, \quad (12.122)$$

with

$$\omega_0 = \pi \sqrt{\frac{2}{\ln 2}},$$

where  $\omega_0$  is such that the second maximum of  $\Re(\psi(t))$  is half of the first one (at  $t = 0$ ), and the scale factor  $1/\sqrt{2\pi}$  makes the wavelet of unit norm. It is to be noted that  $\Psi(0) \neq 0$ , and as such the wavelet is not admissible. However,  $\Psi(0)$  is very small (of order  $10^{-7}$ ) and has numerically no consequence (and can be corrected by removing it from the wavelet). Figure 12.44 shows the Morlet wavelet in time and frequency domains.

It is interesting to note that the Morlet wavelet and the Gabor function are

related. From (12.122) the Morlet wavelet at scale  $a \neq 0$  is

$$\psi_{a,0} = \frac{1}{\sqrt{2\pi a}} e^{-j\omega_0 t/a} e^{-(t/a)^2/2}$$

while, following (11.3) and (11.6), the Gabor function at  $\omega$  is

$$g_{\omega,0}(t) = \frac{1}{\sqrt{2\pi a}} e^{j\omega t/a} e^{-t^2/2a^2}$$

which are equal for  $\omega = \omega_0 = \pi\sqrt{2/\ln 2}$  and the same scale factor  $a$ . Thus, there is a frequency and a scale where the continuous wavelet transform (with a Morlet wavelet) and a local Fourier transform (with a Gabor function) coincide.

## 12.6 Computational Aspects

The multiresolution framework derived above is more than just of theoretical interest. In addition to allow constructing wavelets, like the spline and Meyer wavelets, it also has direct algorithmic implications as we show by deriving Mallat's algorithm for the computation of wavelet series.

### 12.6.1 Wavelet Series: Mallat's Algorithm

Given a wavelet basis  $\{\psi_{m,n}(t)\}_{m,n \in \mathbb{Z}}$ , any function  $x(t)$  can be written as

$$x(t) = \sum_{m \in \mathbb{Z}} \sum_{n \in \mathbb{Z}} \beta_k^{(\ell)} \psi_{m,n}(t).$$

where

$$\beta_k^{(\ell)} = \langle f, \psi_{m,n} \rangle. \quad (12.123)$$

Assume that only a finite-resolution version of  $x(t)$  can be acquired, in particular the projection of  $x(t)$  onto  $V^{(0)}$ , denoted  $f^{(0)}(t)$ . Because

$$V^{(0)} = \bigoplus_{m=1}^{\infty} W^{(\ell)},$$

we can write

$$f^{(0)}(t) = \sum_{m=1}^{\infty} \sum_{n \in \mathbb{Z}} \beta_k^{(\ell)} \psi_{m,n}(t). \quad (12.124)$$

Since  $f^{(0)}(t) \in V^{(0)}$ , we can also write

$$f^{(0)}(t) = \sum_{n \in \mathbb{Z}} \alpha_n^{(0)} \varphi(t-n), \quad (12.125)$$

where

$$\alpha_n^{(0)} = \langle x(t), \varphi(t-n) \rangle_t = \langle f, \varphi_{0,n} \rangle.$$

## 12.6. Computational Aspects

853

Given these two ways of expressing  $f^{(0)}(t)$ , how to go from one to the other? The answer, as to be expected, lies in the two-scale equation, and leads to a filter bank algorithm. Consider  $f^{(1)}(t)$ , the projection of  $f^{(0)}(t)$  onto  $V^{(1)}$ . This involves computing the inner products

$$\alpha_n^{(1)} = \langle f^{(0)}(t), \frac{1}{\sqrt{2}}\varphi(t/2 - n) \rangle_t, \quad n \in \mathbb{Z}. \quad (12.126)$$

From (12.48), we can write

$$\frac{1}{\sqrt{2}}\varphi(t/2 - n) = \sum_{k \in \mathbb{Z}} g_k \varphi(t - 2n - k). \quad (12.127)$$

Replacing this and (12.125) into (12.126) leads to

$$\begin{aligned} \alpha_n^{(1)} &= \sum_{k \in \mathbb{Z}} \sum_{\ell \in \mathbb{Z}} g_k \alpha_\ell^{(0)} \langle \varphi(t - 2n - k), \varphi(t - \ell) \rangle_t \\ &\stackrel{(a)}{=} \sum_{\ell \in \mathbb{Z}} g_{\ell - 2n} \alpha_\ell^{(0)} \stackrel{(b)}{=} (\tilde{g} * \alpha^{(0)})_{2n}, \end{aligned} \quad (12.128)$$

where (a) follows because the inner product is 0 unless  $\ell = 2n + k$ ; and (b) simply rewrites the sum as a convolution, with

$$\tilde{g}_n = g_{-n}.$$

The upshot is that the sequence  $\alpha_n^{(1)}$  is obtained from convolving  $\alpha_n^{(0)}$  with  $\tilde{g}$  (the time-reversed impulse response of  $g$ ) and downsampling by 2. The same development for the wavelet series coefficients

$$\beta_k^{(1)} = \langle f^{(0)}(t), \frac{1}{\sqrt{2}}\psi(t/2 - n) \rangle_t$$

yields

$$\beta_k^{(1)} = (\tilde{h} * \alpha^{(0)})_{2n}, \quad (12.129)$$

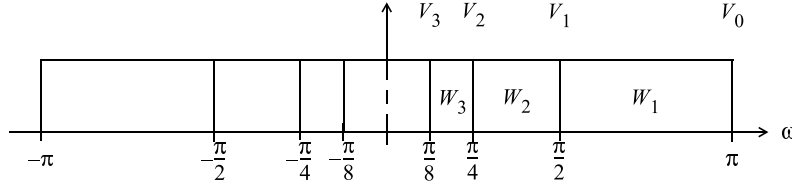
where

$$\tilde{h}_n = h_{-n}$$

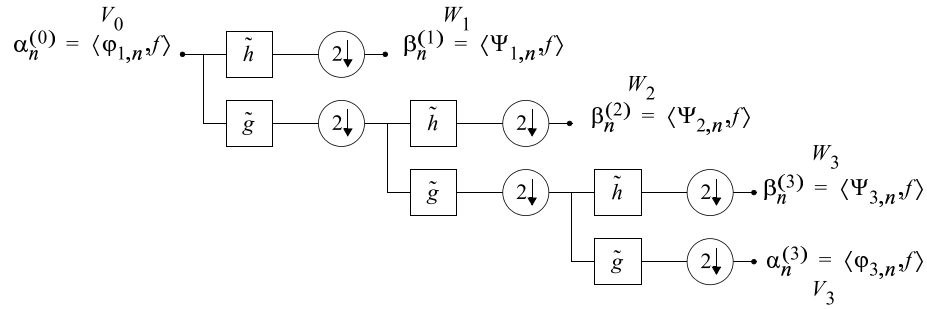
is the time-reversed impulse response of the highpass filter  $h$ . The argument just developed holds irrespectively of the scale at which we start, thus allowing to split a function  $f^{(\ell)}$  in  $V^{(\ell)}$  into its components  $f^{(m+1)}$  in  $V^{(m+1)}$  and  $d^{(m+1)}$  in  $W^{(m+1)}$ . This split is achieved by filtering and downsampling  $\alpha^{(\ell)}$  with  $\tilde{g}$  and  $\tilde{h}$ , respectively. Likewise, this process can be iterated  $k$  times, to go from  $V^{(\ell)}$  to  $V^{(m+k)}$ , while splitting off  $W^{(m+1)}$ ,  $W^{(m+2)}$ ,  $\dots$ ,  $W^{(m+k)}$ , or

$$V^{(\ell)} = W^{(m+1)} \oplus W^{(m+1)} \oplus \dots \oplus W^{(m+k)} \oplus V^{(m+k)}.$$

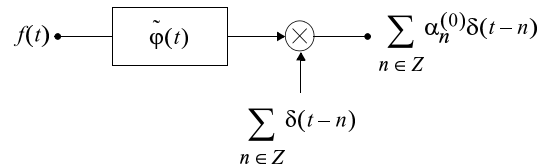
The key insight is of course that once we have an initial projection, for example,  $f^{(0)}(t)$  with expansion coefficients  $\alpha_n^{(0)}$ , then all the other expansion coefficients can be computed using discrete-time filtering. This is shown in Figure 12.46, where the sequence  $\alpha_n^{(0)}$ , corresponding to an initial projection of  $x(t)$  onto  $V^{(0)}$ , is decomposed into the expansion coefficients in  $W^{(1)}$ ,  $W^{(2)}$ ,  $W^{(3)}$  and  $V^{(3)}$ . This algorithm is known as Mallat's algorithm, since it is directly related to the multiresolution analysis of Mallat and Meyer.



**Figure 12.45:** Splitting of  $V^{(0)}$  into  $W^{(1)}, W^{(2)}, W^{(3)}$  and  $V^{(3)}$ , shown for a sinc multiresolution analysis.



**Figure 12.46:** Mallat's algorithm. From the initial sequence  $\alpha_n^{(0)}$ , all of the wavelet series coefficients are computed through a discrete filter bank algorithm.



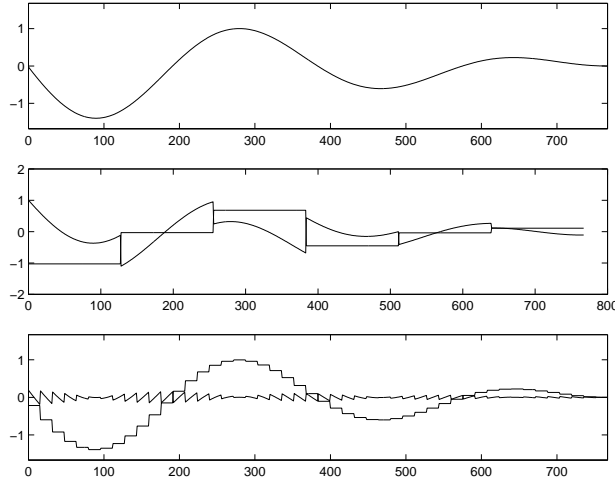
**Figure 12.47:** Initialization of Mallat's algorithm. The function  $x(t)$  is convolved with  $\tilde{\varphi}(t) = \varphi(-t)$  and sampled at  $t = n$ .

**Initialization** How do we initialize Mallat's algorithm, that is, compute the initial sequence  $\alpha_n^{(0)}$ ? There is no escape from computing the inner products

$$\alpha_n^{(0)} = \langle x(t), \varphi(t-n) \rangle_t = (\tilde{\varphi} * f)|_{t=n},$$

where  $\tilde{\varphi}(t) = \varphi(-t)$ . This is shown in Figure 12.47.

The simplification obtained through this algorithm is the following. Computing inner products involves continuous-time filtering and sampling, which is difficult. Instead of having to compute such inner products at all scales as in (12.123), only a single scale has to be computed, namely the one leading to  $\alpha_n^{(0)}$ . All the subsequent inner products are obtained from that sequence, using only discrete-time processing.



**Figure 12.48:** Initial approximation in Mallat's algorithm. (a) Function  $x(t)$ . (b) Approximation  $f^{(0)}(t)$  with Haar scaling function in  $V^{(0)}$  and error  $e^{(0)}(t) = x(t) - f^{(0)}(t)$ . (c) Same but in  $V^{(-3)}$ , or  $f^{(-3)}(t)$  and  $e^{(-3)}(t) = x(t) - f^{(-3)}(t)$ .

The question is: How well does  $f^{(0)}(t)$  approximate the function  $x(t)$ ? The key is that if the error  $\|f^{(0)} - f\|$  is too large, we can go to finer resolutions  $f^{(\ell)}$ ,  $m < 0$ , until  $\|f^{(\ell)} - f\|$  is small enough. Because of completeness, we know that there is an  $m$  such that the initial approximation error can be made as small as we like.

In Figure 12.48, we show two different initial approximations and the resulting errors,

$$e^{(\ell)}(t) = x(t) - f^{(\ell)}(t).$$

Clearly, the smoother the function, the faster the decay of  $\|e^{(\ell)}\|$  as  $m \rightarrow -\infty$ . Exercise 12.13 explores this further.

**The Synthesis Problem** We have considered the analysis problem, or given a function, how to obtain its wavelet coefficients. Conversely, we can also consider the synthesis problem. That is, given a wavelet series representation as in (12.124), how to synthesize  $f^{(0)}(t)$ . One way is to effectively add wavelets at different scales and shifts, with the appropriate weights (12.123).

The other way is to synthesize  $f^{(0)}(t)$  as in (12.125), which now involves only linear combinations of a single function  $\varphi(t)$  and its integer shifts. To make matters specific, assume we want to reconstruct  $f^{(0)} \in V^{(0)}$  from  $f^{(1)}(t) \in V^{(1)}$  and

$d^{(1)}(t) \in W^{(1)}$ . There are two ways to write  $f^{(0)}(t)$ , namely

$$f^{(0)}(t) = \sum_{n \in \mathbb{Z}} \alpha_n^{(0)} \varphi(t - n) \quad (12.130)$$

$$= \frac{1}{\sqrt{2}} \sum_{n \in \mathbb{Z}} \alpha_n^{(1)} \varphi(t/2 - n) + \frac{1}{\sqrt{2}} \sum_{n \in \mathbb{Z}} \beta_k^{(1)} \psi(t/2 - n), \quad (12.131)$$

where the latter is the sum of  $f^{(1)}(t)$  and  $d^{(1)}(t)$ . Now,

$$\alpha_\ell^{(0)} = \langle f^{(0)}(t), \varphi(t - \ell) \rangle_t.$$

Using the two-scale equation (12.127) and its equivalent for  $\psi(t/2 - n)$ ,

$$\frac{1}{\sqrt{2}} \psi(t/2 - n) = \sum_{k \in \mathbb{Z}} h_k \varphi(t - 2n - k),$$

we can write

$$\begin{aligned} \alpha_\ell^{(0)} &= \langle f^{(1)}(t), \varphi(t - \ell) \rangle_t + \langle d^{(1)}(t), \varphi(t - \ell) \rangle_t \\ &\stackrel{(a)}{=} \sum_{n \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} \alpha_n^{(1)} g_k \langle \varphi(t - \ell), \varphi(t - 2n - k) \rangle_t \\ &\quad + \sum_{n \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} \beta_k^{(1)} h_k \langle \varphi(t - \ell), \varphi(t - 2n - k) \rangle_t \\ &\stackrel{(b)}{=} \sum_{n \in \mathbb{Z}} \alpha_n^{(1)} g_{\ell - 2n} + \sum_{n \in \mathbb{Z}} \beta_k^{(1)} h_{\ell - 2n} \end{aligned} \quad (12.132)$$

where (a) follows from (12.131) using the two-scale equation; and (b) is obtained from the orthogonality of the  $\varphi$ s, unless  $k = \ell - 2n$ . The obtained expression for  $\alpha_\ell^{(0)}$  indicates that the two sequences  $\alpha_n^{(1)}$  and  $\beta_k^{(1)}$  are upsampled by 2 before being filtered by  $g$  and  $h$ , respectively. In other words, a two-channel synthesis filter bank produces the coefficients for synthesizing  $f^{(0)}(t)$  according to (12.130). The argument above can be extended to any number of scales and leads to the synthesis version of Mallat's algorithm, shown in Figure 12.49.

Again, the simplification arises since instead of having to use continuous-time wavelets and scaling functions at many scales, only a single continuous-time prototype function is needed. This prototype function is  $\varphi(t)$  and its shifts, or the basis for  $V^{(0)}$ . Because of the inclusion of all the coarser spaces in  $V^{(0)}$ , the result is intuitive, nonetheless it is remarkable that the multiresolution framework leads naturally to a discrete-time filter bank algorithm.

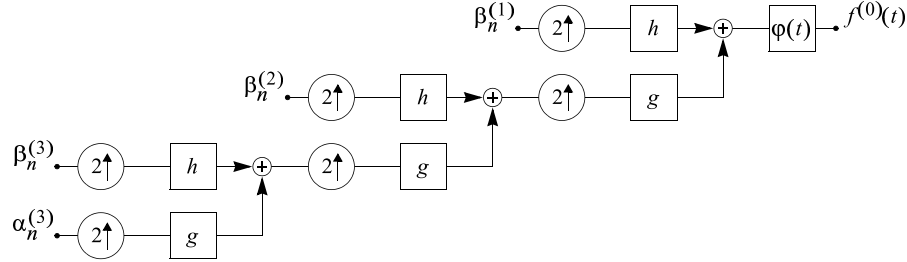
## 12.6.2 Wavelet Frames

### Chapter at a Glance

### Historical Remarks

TBD





**Figure 12.49:** Synthesis of  $f^{(0)}(t)$  using Mallat's algorithm. The wavelet and scaling coefficients are fed through a DWT synthesis, generating the sequence  $\alpha_n^{(0)}$ . A final continuous-time processing implementing (12.130) leads to  $f^{(0)}(t)$ .

	Lowpass & scaling function	Highpass & wavelet
Filter	$G(z) = \left(\frac{1+z^{-1}}{2}\right)^N R(z)$	$H(z) = z^{-L+1} \left(\frac{1-z}{2}\right)^N R(-z^{-1})$
Function	$\Phi(\omega) = \prod_{i=1}^{\infty} \frac{1}{\sqrt{2}} G(e^{j\omega/2^i})$	$\Psi(\omega) = \frac{1}{\sqrt{2}} H(e^{j\omega/2}) \prod_{i=2}^{\infty} \frac{1}{\sqrt{2}} G(e^{j\omega/2^i})$
Two-scale equation	$\Phi(\omega) = 2^{-1/2} G(e^{j\omega/2}) \Phi(\omega/2)$ $\varphi(t) = \sqrt{2} \sum_n g_n \varphi(2t - n)$	$\Psi(\omega) = 2^{-1/2} H(e^{j\omega/2}) \Phi(\omega/2)$ $\psi(t) = \sqrt{2} \sum_n h_n \varphi(2t - n)$
Orthogonality	$\langle \varphi(t), \varphi(t-n) \rangle_t = \delta_n$ $\langle \varphi(t), \psi(t-n) \rangle_t = 0$	$\langle \psi(t), \psi(t-n) \rangle_t = \delta_n$ $\langle \psi(t), 2^{-m/2} \psi(2^{-m}t - n) \rangle_t = \delta_n$
Smoothness	Can be tested, increases with $N$	Same as for $\varphi(t)$
Moments	Polynomials of degree $N-1$ are in $\text{span}(\{\varphi(t-n)\}_{n \in \mathbb{Z}})$	Wavelets has $N$ zero moments
Size and support	$\text{support}(g) = \{0, \dots, L-1\}$ $\text{support}(\varphi) = [0, L-1]$	$\text{support}(h) = \{0, \dots, L-1\}$ $\text{support}(\psi) = [0, L-1]$

**Table 12.1:** Major properties of scaling function and wavelet based on an iterated filter bank with an orthonormal lowpass filter having  $N$  zeros at  $z = -1$  or  $\omega = \pi$ .

## Further Reading

**Books and Textbooks** Daubechies [41].

**Results on Wavelets** For the proof of completeness of Theorem 12.6, see [39, 41].

## Exercises with Solutions

- 12.1. *Proof of Proposition 12.1*  
Prove Proposition 12.1.

*Solution:* The overall strategy of the proof is to show that convergence implies that the sum of the even polyphase component of  $g$ ,  $\sum_n g_{2n}$ , equals the sum of the odd polyphase component,  $\sum_n g_{2n+1}$ . This implies a zero at  $z = -1$ .

To obtain a useful time-domain expression from the  $z$ -domain expression describing the iterative process,  $G^{(J)}(z) = G(z)G^{(J-1)}(z^2)$ , let  $f$  be the sequence with  $z$  transform  $G^{(J-1)}(z^2)$ . ( $f$  is  $g^{(J-1)}$  upsampled by 2.) Then

$$g_n^{(J)} = \sum_{k \in \mathbb{Z}} g_k f_{n-k} = \sum_{\ell \in \mathbb{Z}} g_{2\ell} f_{n-2\ell} + \sum_{\ell \in \mathbb{Z}} g_{2\ell+1} f_{n-(2\ell+1)}, \quad (\text{E12.1-1})$$

by breaking the convolution sum into  $k = 2\ell$  and  $k = 2\ell + 1$  terms. Now since  $f_n = 0$  for all odd values of  $n$ , in the right side of (E12.1-1), the first sum is zero for odd  $n$  and the second sum is zero for even  $n$ . Thus

$$g_{2n}^{(J)} = \sum_{\ell \in \mathbb{Z}} g_{2\ell} f_{2n-2\ell} = \sum_{\ell=0}^{(L/2)-1} g_{2\ell} g_{n-\ell}^{(J-1)}, \quad (\text{E12.1-2a})$$

$$g_{2n+1}^{(J)} = \sum_{\ell \in \mathbb{Z}} g_{2\ell+1} f_{2n-2\ell} = \sum_{\ell=0}^{(L/2)-1} g_{2\ell+1} g_{n-\ell}^{(J-1)}, \quad (\text{E12.1-2b})$$

where we have also used the length of  $g$  to write finite sums.

Now suppose that  $\lim_{J \rightarrow \infty} \varphi^{(J)}(t)$  exists for all  $t$ , and denote the limit by  $\varphi(t)$ . It must also be true that  $\varphi(\tau) \neq 0$  for some  $\tau$ ; otherwise we contradict that  $\|\varphi^{(J)}\| = 1$  for every  $J$ . With  $n_J$  chosen as any function of  $J$  such that  $\lim_{J \rightarrow \infty} 2n_J/2^J = \tau$ , we must have

$$\lim_{J \rightarrow \infty} 2^{J/2} g_{2n_J}^{(J)} = \lim_{J \rightarrow \infty} 2^{J/2} g_{2n_J+1}^{(J)}$$

because both sides equal  $\varphi(\tau)$ . Multiplying by  $2^{J/2}$  and taking limits in (E12.1-2) and then subtracting, we find

$$\sum_{\ell=0}^{(L/2)-1} (g_{2\ell} - g_{2\ell+1}) \left( \lim_{J \rightarrow \infty} 2^{J/2} g_{n_J-\ell}^{(J-1)} \right) = 0.$$

Since the limit above exists equals  $\varphi(\tau) \neq 0$  for every  $\ell \in \{0, 1, (L/2) - 1\}$ , we conclude  $0 = \sum_{\ell \in \mathbb{Z}} (g_{2\ell} - g_{2\ell+1}) = G(-1)$ .

#### 12.2. Proof of Proposition 12.4

Prove Proposition 12.4.

*Solution:* We need to prove that

$$|\Phi(\omega)| < \frac{c}{1 + |\omega|^{(1+\epsilon)}}, \quad \epsilon > 0,$$

which amounts to proving that

$$|\gamma(\omega)| = \prod_{i=1}^{\infty} 2^{-1/2} R(e^{j\omega/2^i}) < c'(1 + |\omega|)^{(N-1-\epsilon)} \quad (\text{E12.2-1})$$

since we have shown that there is a “smoothing term” of order  $1/|\omega|^N$  for large  $\omega$ . Recall that  $R(e^{j\omega})$  is  $2\pi$ -periodic and that  $R(1) = \sqrt{2}$ . Because  $|R(e^{j\omega})| < 2^{N-1/2}$  by assumption, we can find a constant  $\alpha$  such that

$$|R(e^{j\omega})| \leq \sqrt{2}(1 + \alpha|\omega|).$$

Using a Taylor series for the exponential,

$$|R(e^{j\omega})| \leq \sqrt{2}e^{\alpha|\omega|}. \quad (\text{E12.2-2})$$

Consider now  $\gamma(\omega)$  for  $|\omega| \leq 1$ , and let us find an upper bound based on the bound

on  $|R(e^{j\omega})|$ :

$$\begin{aligned} \sup_{|\omega| \leq 1} |\gamma(\omega)| &= \sup_{|\omega| \leq 1} \prod_{k=1}^{\infty} 2^{-1/2} |R(e^{j\omega/2^k})| \\ &\stackrel{(a)}{\leq} \prod_{k=1}^{\infty} e^{\alpha|\omega/2^k|} = e^{\alpha|\omega|(1/2+1/4+\dots)} \\ &\stackrel{(b)}{\leq} e^{\alpha}, \end{aligned} \tag{E12.2-3}$$

where (a) follows from (E12.2-2); and (b) comes from  $|\omega| \leq 1$ .

For  $|\omega| \geq 1$ , there exists a  $J \geq 1$  such that

$$2^{J-1} \leq |\omega| < 2^J.$$

We can then split the infinite product into two parts, namely

$$\prod_{k=1}^{\infty} 2^{-1/2} |R(e^{j\omega/2^k})| = \prod_{k=1}^J 2^{-1/2} |R(e^{j\omega/2^k})| \cdot \prod_{k=1}^{\infty} 2^{-1/2} |R(e^{j\frac{\omega}{2^J 2^k}})|.$$

Because  $|\omega| < 2^J$ , we can bound the second product by  $e^{\alpha}$  according to (E12.2-3). The first product has  $J$  terms and can be bounded by  $2^{-J/2} \cdot B^J$ . Using  $B < 2^{N-1/2}$ , we can upper bound the first product by  $(2^{N-1-\epsilon})^J$ . This leads to

$$\sup_{2^{J-1} \leq |\omega| < 2^J} |\gamma(\omega)| \leq c'' 2^{J(N-1-\epsilon)} \leq c''' (1 + |\omega|)^{N-1-\epsilon},$$

where we used the fact that  $\omega$  is between  $2^{J-1}$  and  $2^J$ . Thus, the growth of  $|\gamma(\omega)|$  is sufficiently slow and therefore  $\Phi(\omega)$  decays faster than  $1/|\omega|$ , proving continuity of  $\varphi(t)$ .

### 12.3. Multiresolution Analysis with a Riesz Basis for $V^{(0)}$

Consider a multiresolution analysis with a Riesz basis  $\{\theta(t-n)\}_{n \in \mathbb{Z}}$  for  $V^{(0)}$ . Then there exists a scaling function  $\varphi(t)$  that satisfies the two-scale equation (12.48) and  $\{\varphi(t-n)\}_{n \in \mathbb{Z}}$  is an orthonormal basis for  $V^{(0)}$ .

*Solution:* Since  $\theta(t)$  and its integer shifts form a Riesz basis,  $\Theta(\omega)$  satisfies

$$0 < \sum_{k \in \mathbb{Z}} |\Theta(\omega + 2\pi k)|^2 < \infty.$$

We create a new function  $\varphi(t)$  with the Fourier transform

$$\Phi(\omega) = \frac{\Theta(\omega)}{\sqrt{\sum_{k \in \mathbb{Z}} |\Theta(\omega + 2\pi k)|^2}}, \tag{E12.3-1}$$

which satisfies

$$\sum_{\ell \in \mathbb{Z}} |\Phi(\omega + 2\pi \ell)|^2 \stackrel{(a)}{=} \frac{1}{\sum_{k \in \mathbb{Z}} |\Theta(\omega + 2\pi k)|^2} \sum_{\ell \in \mathbb{Z}} |\Theta(\omega + 2\pi \ell)|^2 = 1,$$

where in (a) we pulled the denominator in front of the sum since it is  $2\pi$  periodic. According to (3.73d),  $\varphi(t)$  is thus orthogonal to its integer shifts. The fact that it satisfies a two-scale equation was shown in (12.8) for the Haar case, but the argument based on inclusion of  $V^{(0)}$  in  $V^{(-1)}$  is general.

## Exercises

### 12.1. Sinc Function as an Infinite Product

Prove that

$$\frac{\sin t}{t} = \prod_{k=1}^{\infty} \cos\left(\frac{t}{2^k}\right), \tag{P12.1-1}$$

using the following:

$$\lim_{t \rightarrow 0} \frac{\sin t}{t} = 1, \quad \sin t = 2 \sin \frac{t}{2} \cos \frac{t}{2}.$$

The inspiration for this problems comes from [144]. (*Hint*: Use the trigonometric identity recursively.)

### 12.2. Fourier Domain Iteration of Haar Filter

Consider the Fourier-domain iteration of the stretched Haar filter

$$G(z) = \frac{1}{\sqrt{2}}(1 + z^{-3})$$

(i) Verify that

$$\Phi(\omega) = e^{-j3\omega/2} \frac{\sin(3\omega/2)}{3\omega/2}$$

(ii) Verify that each finite iteration is of norm 1, while the limit is not, showing failure of  $\mathcal{L}^2$  convergence.

### 12.3. Multiscale Equation

Based on the two scale equation (12.47), verify the expressions (12.71a) and (12.72a)

$$\Phi(\omega) = 2^{-k/2} G^{(k)}(e^{j\omega/2^k}) \Phi(\omega/2^k),$$

$$\Psi(\omega) = 2^{-k/2} H^{(k)}(e^{j\omega/2^k}) \Phi(\omega/2^k),$$

as well as their time domain equivalents given in (12.71b) and (12.72b).

### 12.4. Scaling Behavior of Wavelet Coefficients around Singularities

In Proposition 12.8, it is stated that wavelet coefficients close to singularities of order  $k$  behave as

$$\beta_n^{(m)} \sim 2^{m(k-1/2)}$$

for  $m \rightarrow -\infty$ . This was shown for  $k = 0$  and 1 in the text. Extend the method used to prove the case  $k = 1$  to include larger  $k$ 's and thus prove (12.84) in general.

### 12.5. Best Least Squares Approximation in the Haar Case

Consider a function  $f_{-1}(t)$  in  $V_{-1}$ , the space of functions constant over half integer intervals. Show that the best least squares approximation  $f_0(t)$  in  $V_0$ , the space of functions constant over integer intervals, is given by the average over two successive intervals.

### 12.6. Two-Scale Equation for Piecewise Linear Spaces

In Example 12.6, we saw various bases for piecewise linear spaces. From the fact that  $\varphi(t)$  satisfies a two scale equation, derive the two scale equation for the orthonormal scaling function  $\varphi(t)$ .

(i) Give the two scale equation for  $\theta(t)$ .

(ii) From the expression for  $\Phi(\omega)$  in (E12.3-1) and the two scale equation for  $\Theta(\omega)$ , derive the two scale equation for  $\Phi(\omega)$ .

(iii) Derive the expression for  $G(e^{j\omega})$ , the Fourier transform of the sequence of coefficients of the two scale equation.

*Note* This can be done for either the causal case,  $\varphi_c(t)$ , or the symmetric case,  $\varphi_s(t)$ .

### 12.7. Wavelets for Piecewise Linear Spaces

Given the coefficients of the two scale equation for the orthonormal scaling function  $\varphi(t)$  (see Exercise 12.6), derive an expression for the wavelet, based on the Fourier expression

$$\Psi(\omega) = -\frac{1}{\sqrt{2}} e^{-j\omega/2} G^*(e^{j(\omega/2+\pi)}) \cdot \Phi(\omega/2)$$

where  $G(e^{j\omega})$  is the discrete Fourier transform of the coefficients of the two scale equation for  $\varphi(t)$ .

12.8. *Sinc Multiresolution Analysis*

Consider the sequence of spaces of bandlimited functions

$$V_m = BL([-2^{-m}\pi, 2^{-m}\pi])$$

with an orthonormal basis for  $V_0$  given by

$$\varphi(t) = \frac{\sin(\pi t)}{\pi t}$$

and its integer shifts.

- (i) Verify that the axioms of multiresolution analysis in Definition 12.10 are satisfied.
- (ii) Given the embedding  $V_0 \subset V_{-1}$ , derive the two-scale equations coefficients  $g_n$  in

$$\varphi(t) = \sqrt{2} \sum_{n \in \mathbb{Z}} g_n \varphi(2t - n).$$

- (iii) Derive the wavelet based on the highpass filter<sup>162</sup> and give its expressions in both time and frequency domains.
- (iv) Verify that the wavelet spaces are

$$W_m = BL([-2^{-m+1}\pi, -2^{-m}\pi] \cup [2^{-m}\pi, 2^{-m+1}\pi]).$$

12.9. *Meyer Scaling Function and Wavelet*

In Example 12.7, we derived one of the simplest Meyer wavelet, based on a continuous  $\Phi(\omega)$ . We generalize this to smoother  $\Phi(\omega)$ 's. For this purpose, introduce a helper function  $a(x)$  that is 0 for  $x \leq 0$  and 1 for  $x \geq 1$ , and satisfies

$$a(x) + a(1-x) = 1 \quad \text{for } 0 \leq x \leq 1. \quad (\text{P12.9-1})$$

An example of such a function is

$$a(x) = \begin{cases} 0 & x \leq 0 \\ 3x^2 - 2x^3 & 0 \leq x \leq 1 \\ 1 & x \geq 1. \end{cases} \quad (\text{P12.9-2})$$

Construct the scaling function  $\Phi(\omega)$  as

$$\Phi(\omega) = \sqrt{a(2 - \frac{3|\omega|}{2\pi})} \quad (\text{P12.9-3})$$

- (i) Verify that  $a(x)$  in (P12.9-2) satisfies (P12.9-1) and that it has a continuous first derivative.
- (ii) Verify that  $\Phi(\omega)$  given by (P12.9-3) satisfies

$$\sum_{k \in \mathbb{Z}} |\Phi(\omega + 2\pi k)|^2 = 1$$

and thus, that  $\{\varphi(t-n)\}_{n \in \mathbb{Z}}$  is an orthonormal set. *Hint:* start by using  $a(t)$  given in (P12.9-2), and then a general  $a(t)$  as in (P12.9-1).

- (iii) With  $V_0 = \text{span}(\{\varphi(t-n)\}_{n \in \mathbb{Z}})$  and  $V_m$  defined the usual way, prove that

$$V_0 \subset V_{-1}$$

- (iv) Show that there exists a  $2\pi$ -periodic function  $G(e^{j\omega})$  such that

$$\Phi(\omega) = \frac{1}{\sqrt{2}} G(e^{j\omega/2}) \Phi(\omega/2) \quad (\text{P12.9-4})$$

and that

$$G(e^{j\omega}) = \sqrt{2} \sum_{k \in \mathbb{Z}} \Phi(2\omega + 4\pi k) \quad (\text{P12.9-5})$$

<sup>162</sup>This differs from TBD in that we skip the shift by  $L$ , since here we have a two-sided infinite filter impulse response.

- (v) Verify (12.86b) by showing that

$$\langle f, \varphi_{m,n} \rangle = 0, \quad m, n \in \mathbb{Z}$$

implies necessarily that  $f = 0$ .

- (vi) Verify (12.86c) by showing that if

$$f \in \bigcap_{m \in \mathbb{Z}} V_m$$

then necessarily  $f = 0$ .

- (vii) From (P12.9-5) and the usual construction of the wavelet, give an expression for  $\Psi(\omega)$  in terms of  $\Phi(\omega)$ .
- (viii) For  $a(x)$  given in (P12.9-2), what decay is expected for  $\varphi(t)$  and  $\psi(t)$ ?
- 12.10. *Admissibility of Daubechies Wavelets*  
Show that all orthonormal and compactly supported wavelets from the Daubechies family satisfy the admissibility condition (12.101).

- 12.11.
- Finite-Dimensional Reproducing Kernels*

Consider an  $M$ -by- $N$  matrix  $T$ ,  $M > N$ , that maps vectors  $x$  from  $\mathbb{R}^N$  into vectors  $y$  living on a subspace  $S$  of  $\mathbb{R}^M$ .

- (i) For
- $T$
- having orthonormal columns, or
- $T^T T = I$
- , and
- $K = T T^T$
- (see (12.113)), what can you say about the vector

$$\hat{y} = Ky,$$

where  $y$  is an arbitrary vector from  $\mathbb{R}^M$ ?

- (ii) For  $T$  having  $N$  linearly independent (but not necessary orthonormal) columns, give a simple test to check whether a vector  $y$  in  $\mathbb{R}^M$  belongs to  $S$  (see (12.114)).
- (iii) In case (ii) above, indicate how to compute the orthogonal projection of an arbitrary vector  $y$  in  $\mathbb{R}^M$  onto  $S$ .
- 12.12. *Reproducing Kernel Formula for the Wavelet Transform*  
Show the converse part of Proposition 12.15. That is, show that if a function  $F(a, b)$  satisfies (12.116), then there exists a function  $f(t)$  with the wavelet transform equal to  $F(a, b)$ .
- 12.13. *Initialization of Mallat's Algorithm*  
Create an approximation problem for a smooth function (for example, bounded with bounded derivative) and compare rate of decay for Haar and piecewise linear approximation. Details later.

- 12.14.
- Biorthogonal Multiresolution Analysis*

Consider the hat function

$$\varphi(t) = \begin{cases} 1 - |t| & |t| \leq 1 \\ 0 & \text{else} \end{cases}$$

and the family  $\{\varphi(t - n)\}_{n \in \mathbb{Z}}$ .

- (i) Characterize  $V_0 = \text{span}(\{\varphi(t - n)\}_{n \in \mathbb{Z}})$
- (ii) Evaluate the deterministic autocorrelation sequence

$$a_n = \langle \varphi(t), \varphi(t - n) \rangle$$

verifying that  $\varphi(t)$  is not orthogonal to its integer translates.

- (iii) Define the usual scaled versions of  $V_0, V_m$ . Verify that the axioms of multiresolution analysis given in Section 12.3.3.
- 12.15. *Geometry of Biorthogonal Multiresolution Analysis*  
Consider the biorthogonal family  $\{\varphi(t), \tilde{\varphi}(t), \psi(t), \tilde{\psi}(t)\}$  as defined in (12.68b), (12.69b), TBD and TBD, as well as the associated multiresolution spaces  $\{V_m, \tilde{V}_m, W_m, \tilde{W}_m\}$ .

- (i) Verify that

$$V_m = V_{m+1} \oplus W_{m+1}$$

and similarly for  $\tilde{V}_m$ .

- (ii) Show that  $V_m$  and  $W_m$  are not orthogonal to each other.  
 (iii) Verify the orthogonality relations

$$\tilde{W}_m \perp V_m$$

and

$$W_m \perp \tilde{V}_m$$

- (iv) Show further that

$$\tilde{W}_m \perp W_{m+k} \quad k \neq 0$$

*Hint:* Show this first for  $k = 1, 2, \dots$  using part 1.





## Chapter 13

# Approximation, Estimation, and Compression

## 13.1 Introduction

### Chapter Outline

## 13.2 Abstract Models and Approximation

### 13.2.1 Local Fourier and Wavelet Approximations of Piecewise Smooth Functions

### 13.2.2 Wide-Sense Stationary Gaussian Processes

### 13.2.3 Poisson Processes

## 13.3 Empirical Models

### 13.3.1 $\ell^p$ Models

### 13.3.2 Statistical Models

## 13.4 Estimation and Denoising

### 13.4.1 Connections to Approximation

### 13.4.2 Wavelet Thresholding and Variants

### 13.4.3 Frames

## 13.5 Compression

### 13.5.1 Audio Compression

### 13.5.2 Image Compression

## 13.6 Inverse Problems

### 13.6.1 Deconvolution

### 13.6.2 Compressed Sensing

### Chapter at a Glance

TBD

### Historical Remarks

TBD

### Further Reading

TBD

---

**Appendix****13.A Elements of Source Coding****13.A.1 Entropy Coding****13.A.2 Quantization****13.A.3 Transform Coding**



# Bibliography

- [1] T. Aach. New criteria for shift variance and wide-sense cyclostationarity in multirate filter banks. In *Proc. IEEE Int. Workshop on Spectral Methods and Multirate Sign. Proc.*, pages 7–13, Florence, Italy, 2006.
- [2] T. Aach. Comparative analysis of shift variance and cyclostationarity in multirate filter banks. *IEEE Trans. Circ. and Syst.*, 54(5):1077–1087, May 2007.
- [3] N. I. Akhiezer and I. M. Glazman. *Theory of Linear Operators in Hilbert Spaces*, volume 1. Frederick Ungar Publishing, 1966.
- [4] S. Akkarakaran and P. P. Vaidyanathan. Bifrequency and bispectrum maps: A new look at multirate systems with stochastic inputs. *IEEE Trans. Signal Proc.*, 48(3):723–736, Mar. 2000.
- [5] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image coding using wavelet transform. *IEEE Trans. Image Proc.*, 1(2):205–220, April 1992.
- [6] K. E. Atkinson. *An Introduction to Numerical Analysis*. John Wiley & Sons, New York, NY, second edition, 1989.
- [7] P. Auscher. *Wavelets: Mathematics and Applications*, chapter Remarks on the local Fourier bases, pages 203–218. CRC Press, 1994.
- [8] M. G. Bellanger and J. L. Daguet. TDM-FDM transmultiplexer: Digital polyphase and FFT. *IEEE Trans. Commun.*, 22(9):1199–1204, September 1974.
- [9] J. J. Benedetto and M. C. Fickus. Finite normalized tight frames. *Adv. Comp. Math., sp. iss. Frames*, 18:357–385, 2003.
- [10] T. Berger. *Rate Distortion Theory*. Prentice Hall, Englewood Cliffs, NJ, 1971.
- [11] D. P. Bertsekas and J. N. Tsitsiklis. *Introduction to Probability*. Athena Scientific, Belmont, MA, 2002.
- [12] R. E. Blahut. *Fast Algorithms for Digital Signal Processing*. Addison-Wesley, Reading, MA, 1985. Reprinted with corrections 1987.

- [13] B. G. Bodmann, P. G. Casazza, and G. Kutyniok. A quantitative notion of redundancy for finite frames. In *Journ. Appl. and Comput. Harmonic Analysis*, 2010. To appear.
- [14] H. Bölcskei and F. Hlawatsch. *Gabor Analysis and Algorithms: Theory and Applications*, chapter Oversampled modulated filter banks, pages 295–322. Birkhäuser, Boston, MA, 1998.
- [15] H. Bölcskei and F. Hlawatsch. Oversampled cosine modulated filter banks with perfect reconstruction. *IEEE Trans. Circ. and Syst. II: Analog and Digital Signal Proc.*, 45(8):1057–1071, August 1998.
- [16] H. Bölcskei, F. Hlawatsch, and H. G. Feichtinger. Frame-theoretic analysis of oversampled filter banks. *IEEE Trans. Signal Proc.*, 46(12):3256–3269, December 1998.
- [17] R. N. Bracewell. *The Fourier Transform and Its Applications*. McGraw-Hill, New York, NY, second edition, 1986.
- [18] A. P. Bradley. Shift invariance in the discrete wavelet transform. In *Proc. Digit. Image Comp.*, December 2003.
- [19] P. Brémaud. *Mathematical Principles of Signal Processing: Fourier and Wavelet Analysis*. Springer, 2002.
- [20] A. A. M. L. Bruckens and A. W. M. van den Enden. New networks for perfect inversion and perfect reconstruction. *IEEE Journ. Sel. Areas in Commun.*, 10(1), September 1992.
- [21] P. J. Burt and E. H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Trans. Commun.*, 31(4):532–540, April 1983.
- [22] L. C. Calvez and P. Vilbé. On the uncertainty principle in discrete signals. *IEEE Trans. Circ. and Syst. II: Analog and Digital Signal Proc.*, 39(6):394–395, June 1992.
- [23] E. J. Candès. Curvelet Web Site. <http://www.curvelet.org/papers.html>.
- [24] E. J. Candès, L. Demanet, D. L. Donoho, and L. Ying. Fast discrete curvelet transforms. *Preprint*, 2005.
- [25] P. G. Casazza. The art of frame theory. *Taiwanese Journ. Math.*, 4(2):129–202, 2000.
- [26] P. G. Casazza and J. Kovačević. Equal-norm tight frames with erasures. *Adv. Comp. Math., sp. iss. Frames*, 18:387–430, 2002.
- [27] P. G. Casazza and G. Kutyniok. A generalization of Gram-Schmidt orthogonalization generating all Parseval frames. *Adv. Comp. Math.*, 27:65–78, 2007. Preprint.

- [28] A. Chebira and J. Kovačević. Lapped tight frame transforms. In *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, volume III, pages 857–860, Honolulu, HI, April 2007.
- [29] O. Christensen. *An Introduction to Frames and Riesz Bases*. Birkhäuser, 2002.
- [30] A. Cohen, I. Daubechies, and J.-C. Feauveau. Biorthogonal bases of compactly supported wavelets. *Commun. Pure and Appl. Math.*, 45:485–560, 1992.
- [31] R. R. Coifman, Y. Meyer, S. Quake, and M. V. Wickerhauser. Signal processing and compression with wavelet packets. Technical report, Yale Univ., 1991.
- [32] J. W. Cooley and J. W. Tukey. An algorithm for the machine calculation of complex Fourier series. *Math. of Comput.*, 19:297–301, April 1965.
- [33] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, New York, NY, 1991.
- [34] R. E. Crochiere and L. R. Rabiner. *Multirate Digital Signal Processing*. Prentice Hall, Englewood Cliffs, NJ, 1983.
- [35] A. L. Cunha, J. Zhou, and M. N. Do. The nonsubsampling contourlet transform: Theory, design, and applications. *IEEE Trans. Image Proc.*, 15(10):3089–3101, October 2006.
- [36] Z. Cvetković. Oversampled modulated filter banks and tight Gabor frames in  $\ell^2(\mathbb{Z})$ . In *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, pages 1456–1459, Detroit, MI, May 1995.
- [37] Z. Cvetković and M. Vetterli. Oversampled filter banks. *IEEE Trans. Signal Proc.*, 46(5):1245–1255, May 1998.
- [38] Z. Cvetković and M. Vetterli. Tight Weyl-Heisenberg frames in  $\ell^2(\mathbb{Z})$ . *IEEE Trans. Signal Proc.*, 46(5):1256–1259, May 1998.
- [39] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Commun. Pure and Appl. Math.*, 41:909–996, November 1988.
- [40] I. Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Trans. Inform. Th.*, 36(5):961–1005, September 1990.
- [41] I. Daubechies. *Ten Lectures on Wavelets*. SIAM, Philadelphia, PA, 1992.
- [42] I. Daubechies, A. Grossman, and Y. Meyer. Painless nonorthogonal expansions. *Journ. Math. Phys.*, 27:1271–1283, November 1986.
- [43] M. N. Do and M. Vetterli. The finite ridgelet transform for image representation. *IEEE Trans. Image Proc.*, 12(1):16–28, January 2003.

- [44] M. N. Do and M. Vetterli. The contourlet transform: An efficient directional multiresolution image representation. *IEEE Trans. Image Proc.*, 14(12):2091–2106, December 2005.
- [45] D. L. Donoho and X. Huo. Beamlets and multiscale image analysis. *Preprint*, 2001.
- [46] D. L. Donoho and P. B. Stark. Uncertainty principles and signal recovery. *SIAM J. Appl. Math.*, 49(3):906–931, June 1989.
- [47] E. Dubois. The sampling and reconstruction of time-varying imagery with application in video systems. *Proc. IEEE*, 73(4):502–522, April 1985.
- [48] D. E. Dudgeon and R. M. Mersereau. *Multidimensional Digital Signal Processing*. Prentice Hall, Englewood Cliffs, NJ, 1984.
- [49] H. W. Dudley. The vocoder. *Bell Lab. Rec.*, 18:122–126, December 1939.
- [50] R. J. Duffin and A. C. Schaeffer. A class of nonharmonic Fourier series. *Trans. Amer. Math. Soc.*, 72:341–366, 1952.
- [51] Y. Eldar and H. Bölcskei. Geometrically uniform frames. *IEEE Trans. Inform. Th.*, 49(4):993–1006, April 2003.
- [52] D. Esteban and C. Galand. Applications of quadrature mirror filters to split band voice coding schemes. In *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, pages 191–195, 1995.
- [53] H. G. Feichtinger and T. Strohmer, editors. *Gabor Analysis and Algorithms: Theory and Applications*. Birkhäuser, Boston, MA, 1998.
- [54] G. B. Folland. *A Course in Abstract Harmonic Analysis*. CRC Press, London, UK, 1995.
- [55] D. Gabor. Theory of communication. *Journ. IEE*, 93:429–457, 1946.
- [56] F. R. Gantmacher. *The Theory of Matrices*, volume 1,2. Chelsea Publishing Company, New York, NY, 1959.
- [57] A. Gersho. Asymptotically optimal block quantization. *IEEE Trans. Inform. Th.*, IT-25(4):373–380, July 1979.
- [58] A. Gersho and R. M. Gray. *Vector Quantization and Signal Compression*. Kluwer Academic Publishers, Norwell, MA, 1992.
- [59] I. Gohberg and S. Goldberg. *Basic Operator Theory*. Birkhäuser, Boston, MA, 1981.
- [60] V. K. Goyal. *Single and Multiple Description Transform Coding with Bases and Frames*. SIAM, Philadelphia, PA. In preparation.



- [61] V. K. Goyal. Theoretical foundations of transform coding. *IEEE Signal Proc. Mag.*, 18(5):9–21, September 2001.
- [62] V. K. Goyal, M. Vetterli, and N. T. Thao. Quantized overcomplete expansions in  $\mathbb{R}^N$ : Analysis, synthesis, and algorithms. *IEEE Trans. Inform. Th.*, 44(1):16–31, January 1998.
- [63] E. Grant. *A source book in medieval science*. Harvard Univ. Press, Cambridge, MA, 1974.
- [64] R. M. Gray and D. L. Neuhoff. Quantization. *IEEE Trans. Inform. Th.*, 44(6):2325–2383, October 1998.
- [65] G. R. Grimmett and D. R. Stirzaker. *Probability and Random Processes*. Oxford Univ. Press, second edition, 1992.
- [66] R. W. Hamming. Mathematics on a distant planet. *Amer. Math. Monthly*, 105(7):640–650, August-September 1998.
- [67] D. Han and D. R. Larson. *Frames, bases and group representations*. Number 697 in Memoirs AMS. AMS Press, Providence, RI, 2000.
- [68] T. L. Heath and Euclid. *The Thirteen Books of Euclid's Elements*. Dover Publications, 1956.
- [69] C. Heil and D. Walnut. Continuous and discrete wavelet transforms. *SIAM Review*, 31:628–666, 1989.
- [70] W. Heisenberg. Über den anschaulichen Inhalt der quantentheoretischen Kinetik und Mechanik. *Zeitschrift für Physik*, 43:172–198, 1927.
- [71] C. Herley and M. Vetterli. Biorthogonal bases of symmetric compactly supported wavelets. In M. Farge and et al., editors, *Proc. Wavelets, Fractals and Fourier Transforms*. Oxford Univ. Press, 1991.
- [72] C. Herley and M. Vetterli. Wavelets and recursive filter banks. *IEEE Trans. Signal Proc.*, August 1993.
- [73] C. Herley and M. Vetterli. Orthogonal time-varying filter banks and wavelet packets. *IEEE Trans. Signal Proc.*, 42(10):2650–2663, October 1994.
- [74] O. Herrmann. On the approximation problem in nonrecursive digital filter design. *IEEE Trans. Circ. Theory*, 18:411–413, 1971.
- [75] R. B. Holmes and V. I. Paulsen. Optimal frames for erasures. *Linear Algebra and Its Appl.*, 377:31–51, 2004.
- [76] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge Univ. Press, 1985. Reprinted with corrections 1987.

- [77] J. J. Y. Huang and P. M. Schultheiss. Block quantization of correlated Gaussian random variables. *IEEE Trans. Commun. Syst.*, CS-11(3):289–296, September 1963.
- [78] D. A. Huffman. A method for the construction of minimum redundancy codes. *Proc. IRE*, 40:1098–1101, September 1952.
- [79] R. Ishii and K. Furukawa. The uncertainty principle in discrete signals. *IEEE Trans. Circ. and Syst.*, CAS-33(10):1032–1034, October 1986.
- [80] A. J. Jerri. The Shannon sampling theorem—its various extensions and applications: A tutorial review. *Proc. IEEE*, 65:1565–1596, November 1977.
- [81] J. D. Johnston. A filter family designed for use in Quadrature Mirror Filter Banks. In *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, pages 291–294, Denver, CO, 1980.
- [82] M. C. Jones. The discrete Gerchberg algorithm. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 34(3):624–626, June 1986.
- [83] T. Kailath. *Linear Systems*. Prentice Hall, Englewood Cliffs, NJ, 1980.
- [84] G. Karlsson and M. Vetterli. Theory of two - dimensional multirate filter banks. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 38(6):925–937, June 1990.
- [85] N. G. Kingsbury. The dual-tree complex wavelet transform: A new efficient tool for image restoration and enhancement. In *Proc. Europ. Sig. Proc. Conf.*, pages 319–322, 1998.
- [86] N. G. Kingsbury. Image processing with complex wavelets. *Phil. Trans. Royal Soc. London A*, September 1999.
- [87] N. G. Kingsbury. Complex wavelets for shift invariant analysis and filtering of signals. *Journ. Appl. and Comput. Harmonic Analysis*, 10(3):234–253, May 2001.
- [88] R. Kongsbruck. *Source coding in sensor networks*. PhD thesis, EPFL, Lausanne, Switzerland, 2009.
- [89] J. Kovačević. *Handbook of Circuits and Filters*, chapter z-Transform. CRC Press, June 1995. Second ed., 2002, third ed., 2006.
- [90] J. Kovačević and A. Chebira. *An Introduction to Frames*. Foundations and Trends in Signal Proc. Now Publishers, 2008.
- [91] J. Kovačević and M. Vetterli. Nonseparable multidimensional perfect reconstruction filter banks and wavelet bases for  $\mathcal{R}^n$ . *IEEE Trans. Inform. Th., sp. iss. Wavelet Transforms and Multiresolution Signal Analysis*, 38(2):533–555, March 1992. Chosen for inclusion in Fundamental Papers in Wavelet Theory.

- [92] H. P. Kramer and M. V. Mathews. A linear coding for transmitting a set of correlated signals. *IRE Trans. Inform. Theory*, 23(3):41–46, September 1956.
- [93] E. Kreyszig. *Introductory Functional Analysis with Applications*. John Wiley & Sons, New York, NY, 1978.
- [94] D. Labate, W-Q. Lim, G. Kutyniok, and G. Weiss. Sparse multidimensional representation using shearlets. In *Proc. SPIE Conf. Wavelet Appl. in Signal and Image Proc.*, pages 254–262, Bellingham, WA, 2005.
- [95] D. J. LeGall and A. Tabatabai. Subband coding of digital images using symmetric short kernel filters and arithmetic coding techniques. In *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, pages 761–765, New York, NY, 1988.
- [96] S. P. Lloyd. A sampling theorem for stationary (wide sense) stochastic processes. *Trans. Amer. Math. Soc.*, 92(1):1–12, July 1959.
- [97] Y. Lu and M. N. Do. The finer directional wavelet transform. In *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, volume 4, pages 573–576, Philadelphia, PA, March 2005.
- [98] D. G. Luenberger. *Optimization by Vector Space Methods*. John Wiley & Sons, New York, NY, 1969.
- [99] S. Mallat. Multiresolution approximations and wavelet orthonormal bases of  $L^2(\mathbb{R})$ . *Trans. Amer. Math. Soc.*, 315:69–87, September 1989.
- [100] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1999.
- [101] H. S. Malvar. *Signal Processing with Lapped Transforms*. Artech House, Norwood, MA, 1992.
- [102] R. J. Marks II. *Introduction to Shannon Sampling and Interpolation Theory*. Springer-Verlag, New York, NY, 1991.
- [103] T. G. Marshall. U-L block-triangular matrix and ladder realizations of subband codes. In *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, volume 3, pages 177–180, 1993.
- [104] J. L. Massey and T. Mittelholzer. *Sequences II: Methods in Communication, Security and Computer Sciences*, chapter Welch’s bound and sequence sets for code-division multiple-access systems, pages 63–78. Springer-Verlag, New York, NY, 1993.
- [105] F. Mintzer. Filters for distortion-free two-band multirate filter banks. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 33(3):626–630, June 1985.
- [106] I. Newton. *Opticks or A Treatise of the Reflections, Refractions, Inflections and Colours of Light*. Royal Society, 1703.

- [107] H. J. Nussbaumer. Complex quadrature mirror filters. In *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, volume II, pages 221–223, Boston, MA, 1983.
- [108] A. V. Oppenheim and R. W. Schaffer. *Discrete-Time Signal Processing*. Prentice Hall, Upper Saddle River, NJ, third edition, 2010.
- [109] A. Papoulis. *The Fourier Integral and its Applications*. McGraw-Hill, New York, NY, 1962.
- [110] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, New York, NY, 1965.
- [111] A. Papoulis. Generalized sampling expansion. *IEEE Trans. Circ. and Syst.*, 24(11):652–654, November 1977.
- [112] A. Papoulis. *Signal Analysis*. McGraw-Hill, New York, NY, 1977.
- [113] B. Porat. *Digital Processing of Random Signals*. Prentice Hall, Englewood Cliffs, NJ, 1994.
- [114] B. Porat. *A Course in Digital Signal Processing*. John Wiley & Sons, New York, NY, 1996.
- [115] P. Prandoni and M. Vetterli. *Signal Processing for Communications*. EPFL Press, Lausanne, Switzerland, 2008.
- [116] M. Püschel and J. M. F. Moura. Algebraic signal processing theory: Foundation and 1-D time. *IEEE Trans. Signal Proc.*, 56(8):3572–3585, 2008.
- [117] K. Ramchandran and M. Vetterli. Best wavelet packet bases in a rate-distortion sense. *IEEE Trans. Image Proc.*, 2(2):160–175, April 1993.
- [118] T. A. Ramstad. IIR filter bank for subband coding of images. In *Proc. IEEE Int. Symp. Circ. and Syst.*, pages 827–830, 1988.
- [119] J. M. Renes, R. Blume-Kohout, A. J. Scot, and C. M. Caves. Symmetric informationally complete quantum measurements. *Journ. Math. Phys.*, 45(6):2171–2180, 2004.
- [120] O. Rioul. A discrete-time multiresolution theory. *IEEE Trans. Signal Proc.*, 41(8):2591–2606, August 1993.
- [121] O. Rioul and P. Duhamel. Fast algorithms for discrete and continuous wavelet transforms. *IEEE Trans. Inform. Th., sp. iss. Wavelet Transforms and Multiresolution Signal Analysis*, 38(2):569–586, March 1992.
- [122] K. A. Ross. *Elementary Analysis: The Theory of Calculus*. Springer-Verlag, New York, NY, 1980.
- [123] N. Saito and R. R. Coifman. Local discriminant bases. In *Proc. SPIE Conf. Vis. Commun. and Image Proc.*, pages 2–14, 1994.

- [124] N. Saito and R. R. Coifman. Local discriminant bases and their applications. *Journ. Math. Imag. Vis.*, 5:337–358, 1995.
- [125] A. Sandryhaila, A. Chebira, C. Milo, J. Kovačević, and M. Püschel. Systematic construction of real lapped tight frame transforms. *IEEE Trans. Signal Proc.*, 58(5):2256–2567, May 2010.
- [126] V. P. Sathe and P. P. Vaidyanathan. Effects of multirate systems on the statistical properties of random signals. *IEEE Trans. Signal Proc.*, 41(1):131–146, January 1993.
- [127] I. W. Selesnick. *Wavelets in Signal and Image Analysis*, chapter The double density DWT. Kluwer Academic Publishers, 2001.
- [128] I. W. Selesnick. The double-density dual-tree DWT. *IEEE Trans. Signal Proc.*, 52(5):1304–1314, May 2004.
- [129] I. Shah and T. A. C. M. Kalker. On ladder structures and linear phase conditions for bi-orthogonal filter banks. In *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, pages 181–184, 1994.
- [130] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journ.*, 27:379–423, July 1948. Continued 27:623–656, October 1948.
- [131] E. P. Simoncelli. Simoncelli Web Site. <http://www.cns.nyu.edu/~eero/steerpyr/>.
- [132] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger. Shiftable multiscale transforms. *IEEE Trans. Inform. Th., sp. iss. Wavelet Transforms and Multiresolution Signal Analysis*, 38(2):587–607, March 1992.
- [133] D. Slepian. Some comments on Fourier analysis, uncertainty and modeling. *SIAM Rev.*, 25(3):379–393, July 1983.
- [134] M. J. T. Smith. *IIR analysis/synthesis systems*, chapter in *Subband Image Coding*. Kluwer Academic Press, Boston, MA, 1991. J. W. Woods ed.
- [135] M. J. T. Smith and T. P. Barnwell III. Exact reconstruction for tree-structured subband coders. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 34(3):431–441, June 1986.
- [136] A. K. Soman, P. P. Vaidyanathan, and T. Q. Nguyen. Linear phase paraunitary filter banks: Theory, factorizations and applications. *IEEE Trans. Signal Proc., sp. iss. Wavelets and Signal Proc.*, 41(12), December 1993.
- [137] J.-L. Starck, M. Elad, and D. L. Donoho. Redundant multiscale transforms and their application for morphological component separation. *Advances in Imaging and Electron Physics*, 132, 2004.
- [138] P. Stoica and R. Moses. *Introduction to Spectral Analysis*. Prentice Hall, Englewood Cliffs, NJ, 2000.

- 
- [139] G. Strang. *Linear Algebra and Its Applications*. Academic Press, New York, NY, 1976.
  - [140] G. Strang. *Linear Algebra and Its Applications*. Brooks/Cole, fourth edition, 2006.
  - [141] G. Strang. *Introduction to Linear Algebra*. <http://math.mit.edu/linearalgebra/>, 4th edition, 2009.
  - [142] G. Strang and G. Fix. *An Analysis of the Finite Element Method*. Wellesley-Cambridge Press, 2nd edition, 2008.
  - [143] G. Strang and T. Nguyen. *Wavelets and Filter Banks*. Wellesley Cambridge Press, Boston, MA, 1996.
  - [144] S. Strogatz. *The Calculus of Friendship: What a Teacher and a Student Learned about Life while Corresponding about Math*. Princeton Univ. Press, 2009.
  - [145] T. Strohmer. *Modern Sampling Theory: Mathematics and Applications*, chapter Finite and infinite-dimensional models for oversampled filter banks, pages 297–320. Birkhäuser, Boston, MA, 2000.
  - [146] T. Strohmer and R. Heath. Grassmannian frames with applications to coding and communications. *Journ. Appl. and Comput. Harmonic Analysis*, 14(3):257–175, 2003.
  - [147] W. Sweldens. The lifting scheme: A custom-design construction of biorthogonal wavelets. *Journ. Appl. and Comput. Harmonic Analysis*, 1996.
  - [148] C. W. Therrien. Issues in multirate statistical signal processing. In *Proc. Asilomar Conf. Sign., Syst. and Comp.*, volume 1, pages 573–576, Pacific Grove, CA, 2001.
  - [149] E. Tolsted. An elementary derivation of Cauchy, Hölder, and Minkowski inequalities from Young’s inequality. *Math. Mag.*, 37(1):2–12, January–February 1964.
  - [150] J. A. Tropp, I. S. Dhillon, R. W. Heath, Jr., and T. Strohmer. Designing structured tight frames via an alternating projection method. *IEEE Trans. Inform. Th.*, 51(1):188–209, January 2005.
  - [151] M. K. Tsatsanis and G. B. Giannakis. Principal component filter banks for optimal multiresolution analysis. *IEEE Trans. Signal Proc.*, 43(8):1766–1777, August 1995.
  - [152] M. Unser. An extension of the Karhunen-Loève transform for wavelets and perfect reconstruction filterbanks. In *Proc. SPIE Conf. Wavelet Appl. in Signal and Image Proc.*, volume 2034, pages 45–56, San Diego, CA, July 1993.

- [153] M. Unser. Wavelets, filterbanks, and the Karhunen-Loève transform. In *Proc. Europ. Conf. Signal Proc.*, volume III, pages 1737–1740, Rhodes, Greece, 1998.
- [154] M. Unser. Splines: A perfect fit for signal and image processing. *IEEE Signal Proc. Mag.*, 16(6):22–38, November 1999.
- [155] M. Unser. Sampling—50 years after Shannon. *Proc. IEEE*, 88(4):569–587, April 2000.
- [156] P. P. Vaidyanathan. Quadrature mirror filter banks, M-band extensions and perfect reconstruction techniques. *IEEE Acoust., Speech, and Signal Proc. Mag.*, 4(3):4–20, July 1987.
- [157] P. P. Vaidyanathan. Theory and design of M-channel maximally decimated quadrature mirror filters with arbitrary M, having the perfect reconstruction property. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 35(4):476–492, April 1987.
- [158] P. P. Vaidyanathan. *Multirate Systems and Filter Banks*. Prentice Hall, Englewood Cliffs, NJ, 1992.
- [159] P. P. Vaidyanathan and S. Akkarakaran. Quantized frame expansions with erasures. *Journ. Appl. and Comput. Harmonic Analysis*, 10(3):254–289, May 2001.
- [160] P. P. Vaidyanathan and P.-Q. Hoang. Lattice structures for optimal design and robust implementation of two-channel perfect reconstruction filter banks. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 36(1):81–94, January 1988.
- [161] P. P. Vaidyanathan and S. K. Mitra. Polyphase networks, block digital filtering, LPTV systems, and alias-free QMF banks: a unified approach based on pseudo-circulants. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 36:381–391, March 1988.
- [162] P. P. Vaidyanathan and Z. Doğanata. The role of lossless systems in modern digital signal processing: A tutorial. *IEEE Trans. Educ.*, 32(3):181–197, August 1989.
- [163] R. Vale and S. Waldron. Tight frames and their symmetries. *Const. Approx.*, 21:83–112, 2005.
- [164] M. Vetterli. Filter banks allowing perfect reconstruction. *Signal Proc.*, 10(3):219–244, April 1986.
- [165] M. Vetterli. A theory of multirate filter banks. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 35(3):356–372, March 1987.
- [166] M. Vetterli and C. Herley. Wavelets and filter banks: Theory and design. *IEEE Trans. Signal Proc.*, 40(9):2207–2232, September 1992.

- 
- [167] M. Vetterli and J. Kovačević. *Wavelets and Subband Coding*. Signal Processing. Prentice Hall, Englewood Cliffs, NJ, 1995. <http://waveletsandsubbandcoding.org/>.
  - [168] M. Vetterli and D. J. LeGall. Perfect reconstruction FIR filter banks: Some properties and factorizations. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 37(7):1057–1071, July 1989.
  - [169] E. Viscito and J. P. Allebach. The analysis and design of multidimensional FIR perfect reconstruction filter banks for arbitrary sampling lattices. *IEEE Trans. Circ. and Syst.*, 38(1):29–42, January 1991.
  - [170] P. Viswanath and V. Anantharam. Optimal sequences and sum capacity of synchronous CDMA systems. *IEEE Trans. Inform. Th.*, 45(6):1984–1991, September 1999.
  - [171] S. Waldron. Generalised Welch bound equality sequences are tight frames. *IEEE Trans. Inform. Th.*, 49(9):2307–2309, September 2003.
  - [172] M. V. Wickerhauser. INRIA lectures on wavelet packet algorithms. Technical report, Yale Univ., March 1991.
  - [173] M. V. Wickerhauser. Lectures on wavelet packet algorithms, 1992.
  - [174] Wikipedia. Hilbert – Wikipedia, the free encyclopedia, 2007.
  - [175] E. Wong and B. Hajek. *Stochastic Processes in Engineering Systems*. Springer Texts in Electrical Engineering. Springer-Verlag, New York, NY, 1985.
  - [176] N. Young. *An Introduction to Hilbert Space*. Cambridge Univ. Press, Cambridge, UK, 1988.